Allen-Bradley

ControlLogix Controllers

1756-L55M12, 1756-L55M13, 1756-L55M14, 1756-L55M16, 1756-L55M22, 1756-L55M23, 1756-L55M24,

1756-L61, 1756-L62, 1756-L63

1756-L60M03SE

Firmware Revision 15

User Manual

Rockwell Automation

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (Publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.ab.com/manuals/gi) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc. is prohibited.

Throughout this manual we use notes to make you aware of safety considerations.

WARNING

Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

IMPORTANT

Identifies information that is critical for successful application and understanding of the product.

ATTENTION

Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you:

- identify a hazard
- avoid a hazard
- recognize the consequence

SHOCK HAZARD

Labels may be located on or inside the drive to alert people that dangerous voltage may be present.

BURN HAZARD

Labels may be located on or inside the drive to alert people that surfaces may be dangerous temperatures.

Developing ControlLogix Controller Systems

Introduction

Use this manual to become familiar with the ControlLogix controller and its features. This version of the manual corresponds to controller firmware revision 15.

This manual describes the necessary tasks to install, configure, program, and operate a ControlLogix system. In some cases, this manual includes references to additional documentation that provides the more comprehensive details.

Related Documentation

These core documents address the Logix5000 family of controllers:

For this information:	Use this publication:
where to start for a new user of a Logix5000 controller program and test a simple project	Logix5000 Controllers Quick Start publication 1756-QS001
how to complete standard tasks program logic using sequential function charts (SFC), ladder diagram (LD), structured text (ST), and function block diagram (FBD) languages	Logix5000 Controllers Common Procedures publication 1756-PM001 **Important:** SFC and ST Programming Languages Programming Manual, publication 1756-PM003, is an excerpt from the Logix5000 Controllers Common Procedures Programming Manual
Logix5000 controller reference: • LED patterns • controller features • instruction set quick reference	Logix5000 Controllers System Reference publication 1756-QR107
program sequential applications ladder diagram and structured text instructions	Logix5000 Controllers General Instruction Set Reference Manual publication 1756-RM003
program process control and drives applications function block diagram instructions	Logix5000 Controllers Process Control/Drives Instruction Set Reference Manual publication 1756-RM006
program motion applications ladder diagram motion instructions	Logix5000 Controllers Motion Instruction Set Reference Manual publication 1756-RM007
configure and program motion interface modules create and configure motion groups and axes configure a coordinated system time master device	Logix5000 Motion Module User Manual publication 1756-Um006

The documents address network communications:

For this information:	Use this publication:
configure and use EtherNet/IP networks communicate over EtherNet/IP	EtherNet/IP Communication Modules in Logix5000 Control Systems publication ENET-UM001
configure and use ControlNet networks communicate over ControlNet	ControlNet Communication Modules in Logix5000 Control Systems publication CNET-UM001
configure and use DeviceNet network communicate over DeviceNet	DeviceNet Communication Modules in Logix5000 Control Systems publication CNET-UM004

These documents address specific controller applications:

For this information:	Use this publication:
adhere to SIL2 requirements	Using ControlLogix in SIL2 Applications Safety Reference Manual publication 1756-RM001
configure and program redundant controller systems	ControlLogix Redundancy System User Manual publication 1756-UM523
use a state model for your controller configure equipment phase programs	PhaseManager User Manual publication LOGIX-UM001

- To view or download manuals, visit www.rockwellautomation.com/literature.
- To obtain a hard copy of a manual, contact your local Rockwell Automation distributor or sales representative.

Chapter 1

Where to Start

Use This Chapter 1-1
Design 1-3
Install Hardware 1-4

Chapter 2

Directly Connect to the Controller via the Serial Port

Use This Chapter 2-1
Connect the Controller to via the Serial Port 2-1
Configure the Serial Driver 2-3
Select the Controller Path 2-5

Chapter 3

Communicate over Networks

Use This Chapter 3-1
EtherNet/IP 3-3
    Connections over EtherNet/IP 3-4
ControlNet 3-5
    Connections over ControlNet 3-7
DeviceNet 3-8
    Connections over DeviceNet 3-9
Serial 3-10
    Communicate with DF1 devices 3-12
    Communicate with ASCII devices 3-14
    Modbus support 3-17
DH-485 3-17
DH+ 3-20
    Communicate over DH+ 3-21
Universal Remote I/O 3-21
    Communicate over universal remote I/O 3-22
FOUNDATION Fieldbus 3-24
HART (Highway Addressable Remote Transducer) Protocol) 3-25

Chapter 4

Manage Controller Communications

Use This Chapter 4-1
Produce and Consume (Interlock) Data 4-1
Send and Receive Messages 4-3
    Determine whether to cache message connections 4-3
Connection Overview 4-4
Calculate Connection Use 4-5
Connections Example 4-6

Chapter 5

Place, Configure, and Monitor I/O

Use This Chapter 5-1
Select I/O Modules 5-1
Place Local I/O Modules 5-2
Configure I/O 5-2
I/O connections 5-4
Configure Distributed I/O on EtherNet/IP 5-5
Configure Distributed I/O on ControlNet 5-6
Configure Distributed I/O on DeviceNet 5-7
Address I/O Data 5-8
Add 1756 I/O at Runtime 5-8
Considerations when adding ControlNet I/O at runtime 5-9
Considerations when adding EtherNet/IP I/O at runtime . . . 5-10
Determine When Data Is Updated 5-11
Reconfigure an I/O Module 5-12
Reconfigure a module via RSLogix 5000 software. 5-12
Reconfigure a module via a MSG instruction 5-13

Chapter 6

Develop Applications

Use This Chapter 6-1
Manage Tasks 6-1
Defining tasks 6-1
Define programs 6-2
Define routines 6-2
Sample controller projects 6-3
Organize Tags. 6-4
Select a Programming Language 6-5
Monitor Controller Status 6-6
Monitor Connections 6-7
Determine if communication has timed out with any device . . . 6-7
Determine if communication has timed out with a specific I/O module 6-8
Interrupt the execution of logic and execute the fault handler . . 6-9

Chapter 7

Configure Motion

Use This Chapter 7-1
Make the Controller the CST Master 7-2
If you have more than 1 controller in the chassis 7-2
Add the Motion Modules. 7-3
Add SERCOS interface Drives 7-4
Set Up Each SERCOS Interface Module 7-5
Add the Motion Group 7-7
Add Your Axes. 7-9
Set Up Each Axis 7-10
Check the Wiring of Each Drive. 7-13
Tune Each Axis 7-14
Program Motion Control. 7-15
Additional Actions 7-17

Chapter 8

Configure PhaseManager

Use This Chapter 8-1
PhaseManager Overview 8-1
State Model Overview 8-3
How equipment changes states. 8-4
Manually change states 8-6
Compare PhaseManager to Other State Models. 8-6
Minimum System Requirements 8-6
Equipment Phase Instructions 8-7

Chapter 9

Configure Redundancy

Use This Chapter 9-1
ControlLogix Redundancy Overview 9-1
Build a Redundant System. 9-3
System considerations 9-4
ControlNet Considerations in Redundant Systems 9-4
EtherNet/IP Considerations in Redundant Systems 9-5
IP address swapping 9-5
Redundancy and Scan Time. 9-6
Minimum System Requirements 9-6

Chapter 10

SIL 2 Certification

Use This Appendix. A-1
SIL 2 Overview A-1
SIL 2 Application A-2

Chapter 11

Maintain Nonvolatile Memory

Use This Chapter B-1
Choose a Controller That Has Nonvolatile Memory B-2
Prevent a major fault during a load B-2
Use a CompactFlash Reader B-3

Chapter 12

Maintain the Battery

Use This Chapter C-1
Check If the Battery Is Low C-2
Estimate 1756-BA1 Battery Life (1756-L55Mx all series and 1756-L6x series A controllers) C-2
Estimate 1756-BA2 Battery Life (1756-L6x series B controllers only) C-4
Estimate warning time C-5
Maintain a 1756-BATM Battery Module (1756-L55Mx all series and 1756-L6x series A controllers only) C-7
Check the BAT LED C-7
Store Batteries C-8

Appendix A

Interpret Controller LEDs

RUN Indicator A-1
I/O Indicator A-1
FORCE Indicator A-1
RS232 Indicator A-2
BAT Indicator A-2
OK Indicator A-2

Appendix B

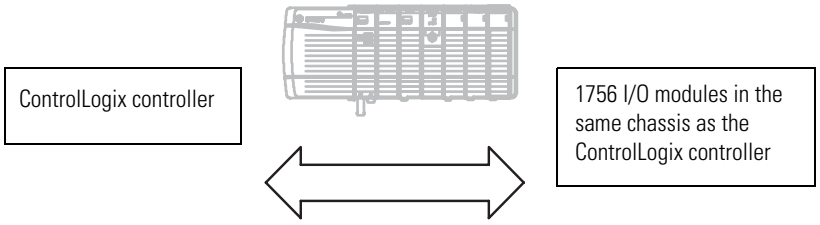
Instruction Locator

Where to Find an Instruction B-1

Where to Start

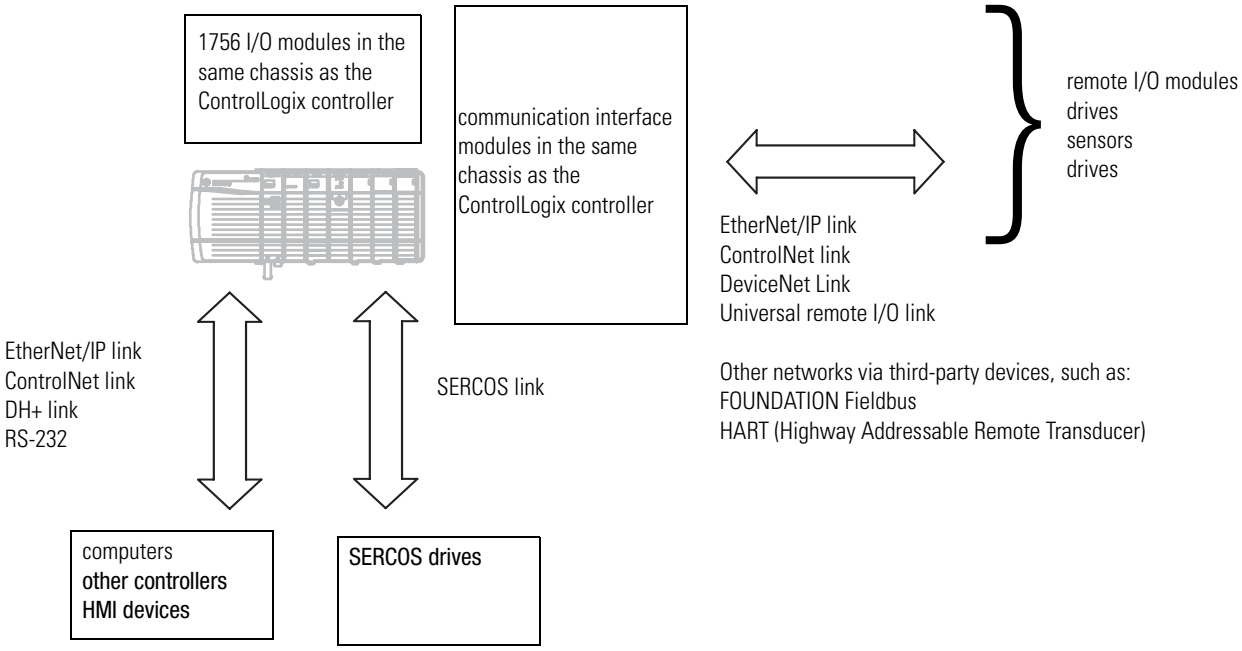
Use This Chapter

The ControlLogix system provides sequential, process, motion, and drive control together with communications and I/O in a chassis-based system. A simple ControlLogix system consists of a stand-alone controller and I/O modules in a single chassis.



For a more flexible system, use:

- multiple controllers in a single chassis
- multiple controllers joined across networks
- I/O from multiple platforms that is distributed in many locations and connected over multiple I/O links



The ControlLogix controller is part of the Logix5000 family of controllers. A ControlLogix system includes:

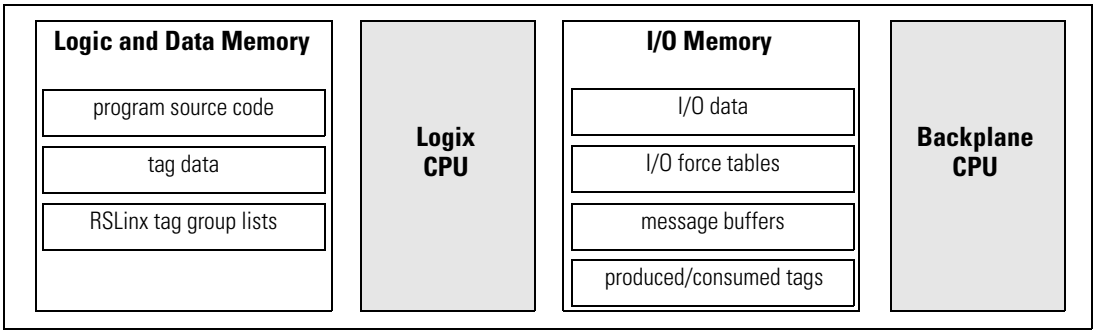
- The ControlLogix controller is available in different combinations of user memory:

This controller:	Memory for:		Nonvolatile Backup Memory:
	Data and Logic	I/O:	
1756-L55M12	750 Kbytes	208 Kbytes	no
1756-L55M13	1.5 Mbytes	208 Kbytes	no
1756-L55M14	3.5 Mbytes	208 Kbytes	no
1756-L55M16	7.5 Mbytes ≤ 3.5 Mbytes of data	208 Kbytes	no
1756-L55M22	750 Kbytes	208 Kbytes	integrated
1756-L55M23	1.5 Kbytes	208 Kbytes	integrated
1756-L55M24	3.5 Kbytes	208 Kbytes	integrated
1756-L61	2 Mbytes	478 Kbytes	CompactFlash(1) card
1756-L62	4 Mbytes	478 Kbytes	CompactFlash(1) card
1756-L63	8 Mbytes	478 Kbytes	CompactFlash(1) card
1756-L60M03SE	750 Kbytes	478 Kbytes	CompactFlash(1) card

(1) CompactFlash is optional and does not come with the controller.

- RSLogix 5000 programming software
- 1756 ControlLogix I/O modules that reside in 1756 chassis
- Different communication modules for EtherNet/IP, ControlNet, DeviceNet, DH+, and Universal remote I/O networks
- Other networks via third-party devices, such as FOUNDATION Fieldbus and HART
- Built-in serial port on every ControlLogix controller

The ControlLogix controller divides resources between a Logix CPU and a backplane CPU:



- The Logix CPU executes application code and messages.
- The backplane CPU communicates with I/O and sends/receives data from the backplane. This CPU operates independently from the Logix CPU, so it sends and receives I/O information asynchronous to program execution.

Design

See:

- *ControlLogix Selection Guide*, 1756-SG001
- *Logix5000 Controller Design Considerations Reference Manual*, 1756-RM094

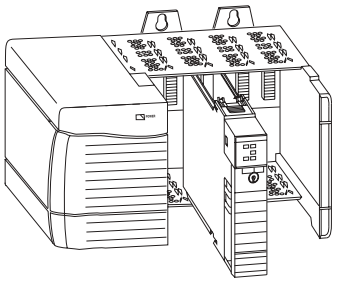
When designing a ControlLogix system, determine the network configuration and the placement of components in each location. Make these decisions as you design your system:

✓	Design Step:
❑ **1.**	Select I/O devices
❑ **2.**	Select motion control and drives requirements
❑ **3.**	Select communication modules
❑ **4.**	Select controllers
❑ **5.**	Select chassis
❑ **6.**	Select power supplies
❑ **7.**	Select software

Install Hardware

See:

- *1756 ControlLogix Controller Installation Instructions*, 1756-IN101



To install a ControlLogix controller, follow these steps:

✓	Installation Step:
❑ **1.**	Install memory options: • on a 1756-L55, install a memory board for additional memory • on a 1756-L6x, install a 1784-CF64 CompactFlash card for nonvolatile memory See Chapter 11 "Maintain Nonvolatile Memory".
❑ **2.**	Connect the battery See Chapter 12 "Maintain the Battery."
❑ **3.**	Install the controller in the chassis
❑ **5.**	Make serial connections See Chapter 2 "Directly Connect to the Controller via the Serial Port"
❑ **6.**	Load controller firmware
❑ **5.**	Make additional network connections See Chapter 3 "Communicate over Networks."

Directly Connect to the Controller via the Serial Port

Use This Chapter

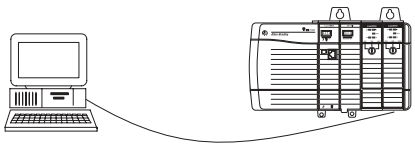
This chapter describes how to connect the controller to the serial port and how to upload/download a project to the controller.

See:

- *EtherNet/IP Modules in Logix5000 Control Systems User Manual*, ENET-UM001
- *ControlNet Modules in Logix5000 Control System User Manual*, CNET-UM001
- *DeviceNet Modules in Logix5000 Control System User Manual*, DNET-UM004

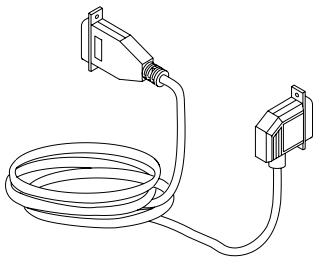
For this information:	See:
Connect the Controller to via the Serial Port	2-1
Configure the Serial Driver	2-3
Select the Controller Path	2-5

Connect the Controller to via the Serial Port



To connect a serial cable:

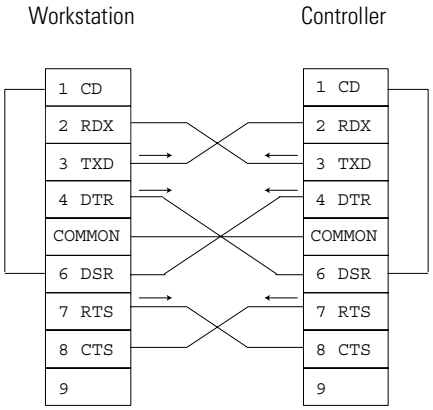
1. Obtain a 1756-CP3 serial cable. (You can also use a 1747-CP3 cable from the SLC product family, but once the cable is connected you cannot close the controller door.)



TIP

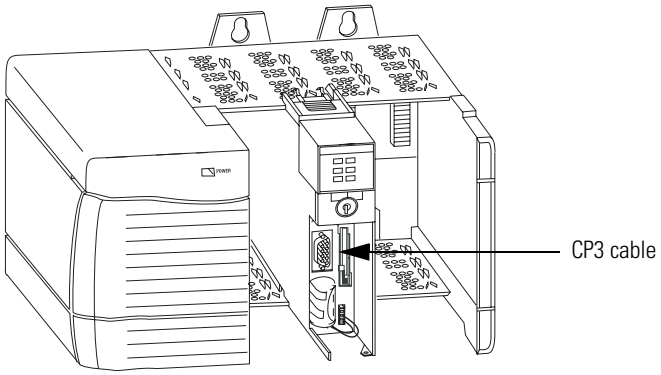
If you make your own serial cable:

- Limit the length to 15.2m (50 ft).
- Wire the connectors as follows:



- Attach the shield to both connectors.

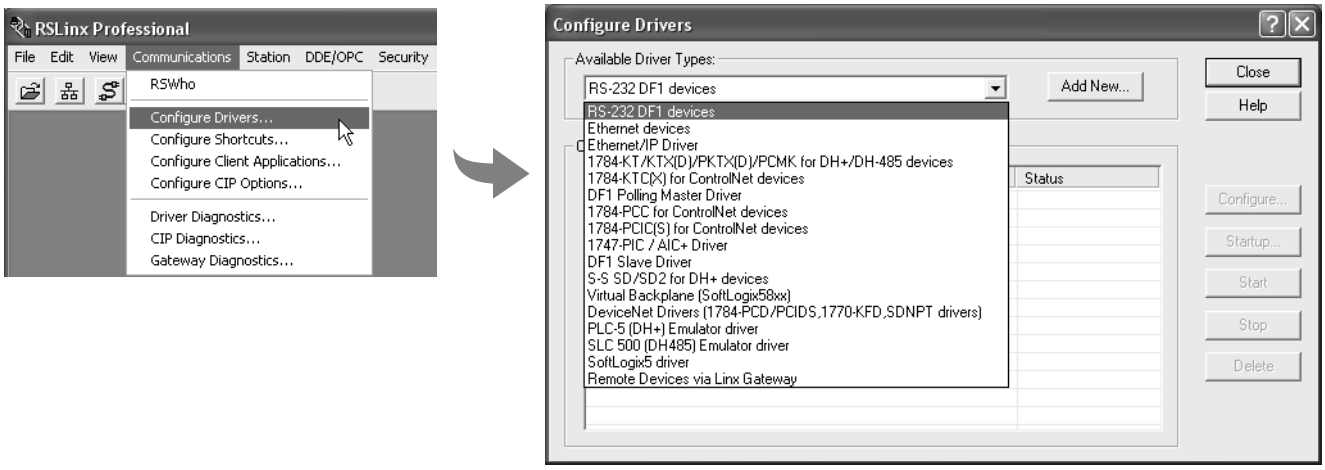
2. Connect the cable to the controller and to your workstation.



Configure the Serial Driver

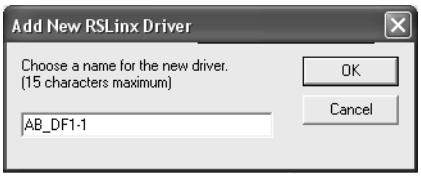
Use RSLinx software to configure the RS-232 DF1 Device driver for serial communications. To configure the driver:

1. From the Communications menu in RSLinx software, select Configure Drivers. Choose the RS-232 DF1 Device driver.



2. Click Add New to add the driver.

3. Specify the driver name and click OK.



4. Specify the serial port settings:
   a. From the Comm Port drop-down list, select the serial port (on the workstation) that the cable is connected to.
   b. From the Device drop-down list, select Logix 5550-Serial Port.
   c. Click Auto-Configure.

5. Does the dialog box display the following message:

   Auto Configuration Successful!

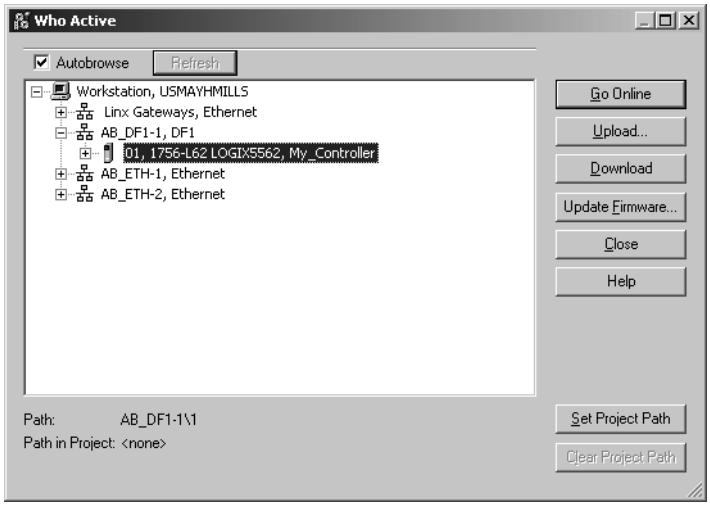
If:	Then:
Yes	Click OK.
No	Go to step 4. and verify that you selected the correct Comm Port.

Then click Close.

Select the Controller Path

In RSLogix 5000 software, select the controller path.

1. Open an RSLogix 5000 project for the controller.
2. From the Communications menu, select Who Active.
3. Expand the communication driver to the level of the controller.



4. Select the controller.

To:	Choose:
monitor the project in the controller	Go Online
transfer a copy of the project from the controller to RSLogix 5000 software	Upload
transfer the open project to the controller	Download

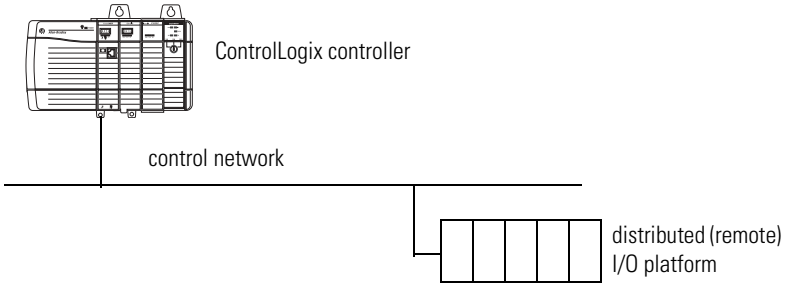
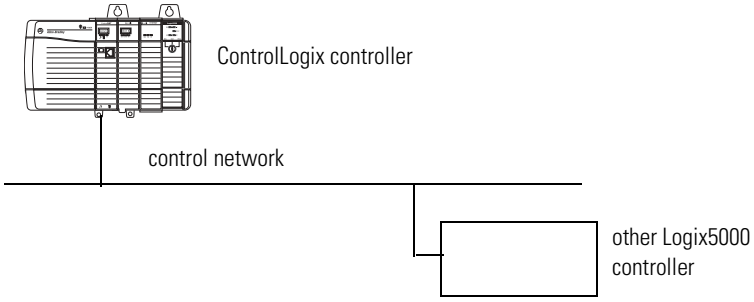
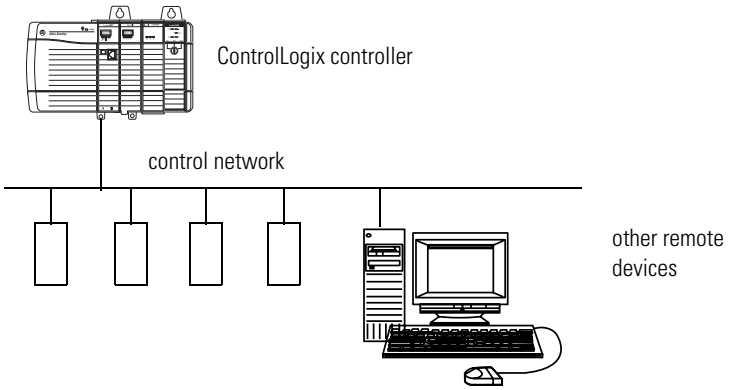
You may have to confirm the action.

Notes:

Communicate over Networks

Use This Chapter

The ControlLogix controller supports additional networks so that the controller can:

Supported networks for:	Example:
Control distributed (remote) I/O • EtherNet/IP • ControlNet • DeviceNet • Universal remote I/O • Foundation Fieldbus • HART	
Produce/consume (interlock) data between controllers • EtherNet/IP • ControlNet	
Send and receive messages to and from other devices (this includes access to the controller via RSLogix 5000 programming software) • EtherNet/IP • ControlNet • DeviceNet (to devices only) • serial • DH+ • DH-485	

This chapter summarizes the ControlLogix controller's communications capabilities:

For this information:	See:
EtherNet/IP	3-3
ControlNet	3-5
DeviceNet	3-8
Serial	3-10
DH-485	3-17
DH+	3-20
Universal Remote I/O	3-21
Foundation Fieldbus	3-24
HART	3-25

EtherNet/IP

See:

- *EtherNet/IP Modules in Logix5000 Control Systems User Manual*, ENET-UM001
- *EtherNet/IP Web Server Module User Manual*, ENET-UM527
- *EtherNet/IP Performance Application Guide*, ENET-AP001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

For EtherNet/IP communications, select from these ControlLogix modules:

If your application:	Select:
• controls I/O modules • requires an adapter for distributed I/O on EtherNet/IP links • communicates with other EtherNet/IP devices (messages) • shares data with other Logix5000 controllers (produce/consume) • bridges EtherNet/IP links to route messages to devices on other networks	1756-ENBT
• requires remote access via Internet browser to tags in a local ControlLogix controller • communicates with other EtherNet/IP devices (messages) • bridges EtherNet/IP links to route messages to devices on other networks • does **NOT** support I/O or produced/consumed tags	1756-EWEB

In addition to communication hardware for EtherNet/IP networks, these software products are available:

- RSLogix 5000 programming software is required

  Use this to configure the ControlLogix project and define EtherNet/IP communications.

- BOOTP/DHCP Utility is optional

  This utility comes with RSLogix 5000 software. Use this utility to assign IP addresses to devices on an EtherNet/IP network.

- RSNetWorx for EtherNet/IP is optional

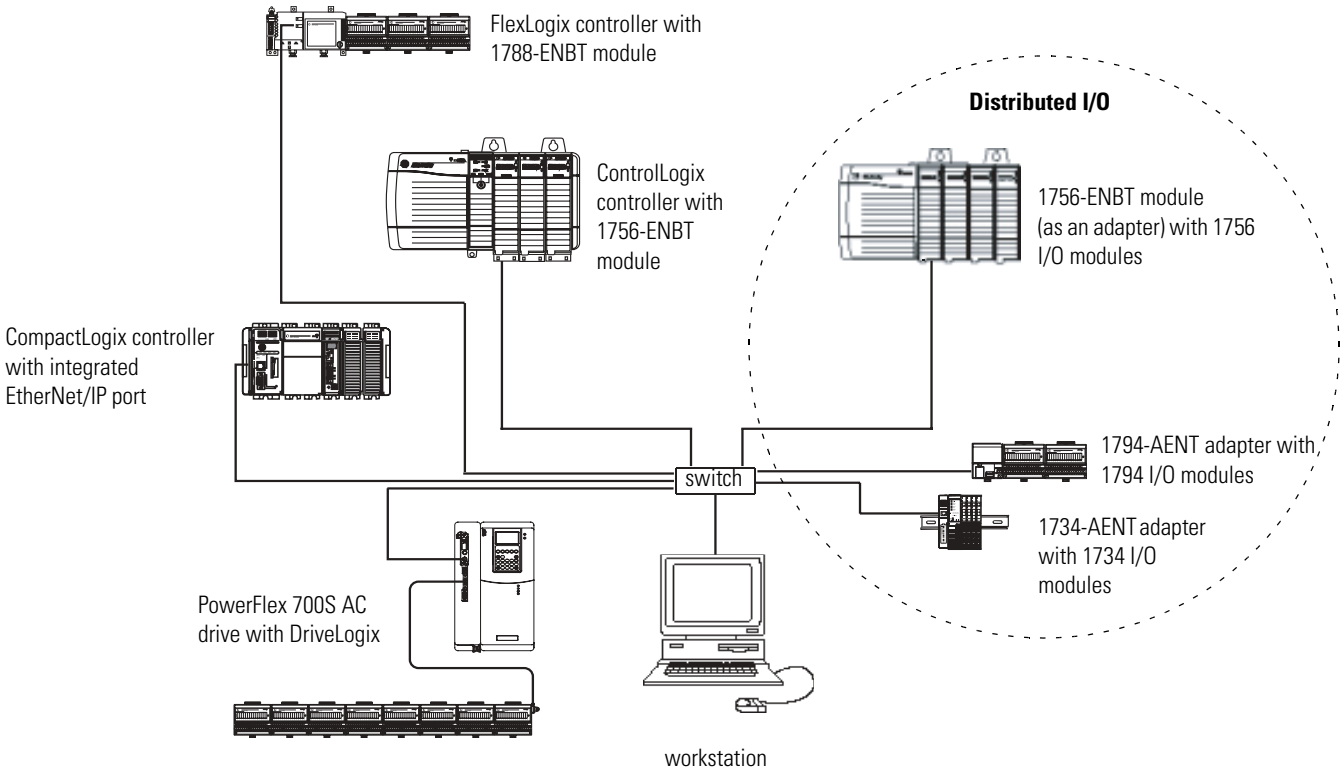
  Use this software to configure EtherNet/IP devices by IP addresses and/or host names.

The EtherNet/IP communication modules:

- support messaging, produced/consumed tags, HMI, and distributed I/O
- encapsulate messages within standard TCP/UDP/IP protocol
- share a common application layer with ControlNet and DeviceNet
- interface via RJ45, category 5, unshielded, twisted-pair cable
- support half/full duplex 10 Mbps or 100 Mbps operation
- support standard switches
- require no network scheduling
- require no routing tables

In this example:

- The controllers can produce and consume tags among each other.
- The controllers can initiate MSG instructions that send/receive data or configure devices.
- The personal computer can upload/download projects to the controllers.
- The personal computer can configure devices on EtherNet/IP.



Connections over EtherNet/IP

You indirectly determine the number of connections the controller uses by configuring the controller to communicate with other devices in the system. Connections are allocations of resources that provide more reliable communications between devices compared to unconnected messages.

All EtherNet/IP connections are unscheduled. An unscheduled connection is triggered by the RPI (requested packet interval for I/O control) or the program (such as a MSG instruction). Unscheduled messaging lets you send and receive data when needed.

The 1756 EtherNet/IP communication modules support 128 CIP (Common Industrial Protocol) connections over an EtherNet/IP network.

For more information... The *EtherNet/IP Modules in Logix5000 Control Systems User Manual*, ENET-UM001 provides information on how to:

- configure an EtherNet/IP communication module
- control I/O over EtherNet/IP
- send a message over EtherNet/IP
- produce/consume a tag over EtherNet/IP
- monitor diagnostics
- calculate controller connections over EtherNet/IP

The *Logix5000 Controllers Design Guidelines Reference Manual*, 1756-RM094 provides guidelines on optimizing a control application on an EtherNet/IP network.

ControlNet

See:

- *Control Net Modules in Logix5000 Control Systems User Manual*, CNET-UM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

For ControlNet communications, select from these ControlLogix modules:

If your application:	Select:
• controls I/O modules • requires an adapter for distributed I/O on ControlNet links • communicates with other ControlNet devices (messages) • shares data with other Logix5000 controllers (produce/consume) • bridges ControlNet links to route messages to devices on other networks	1756-CNB
• performs same functions as a 1756-CNB • also supports redundant ControlNet media	1756-CNBR

In addition to communication hardware for ControlNet networks, these software products are available:

- RSLogix 5000 programming software is required

  Use this to configure the ControlLogix project and define ControlNet communications.

- RSNetWorx for ControlNet is required

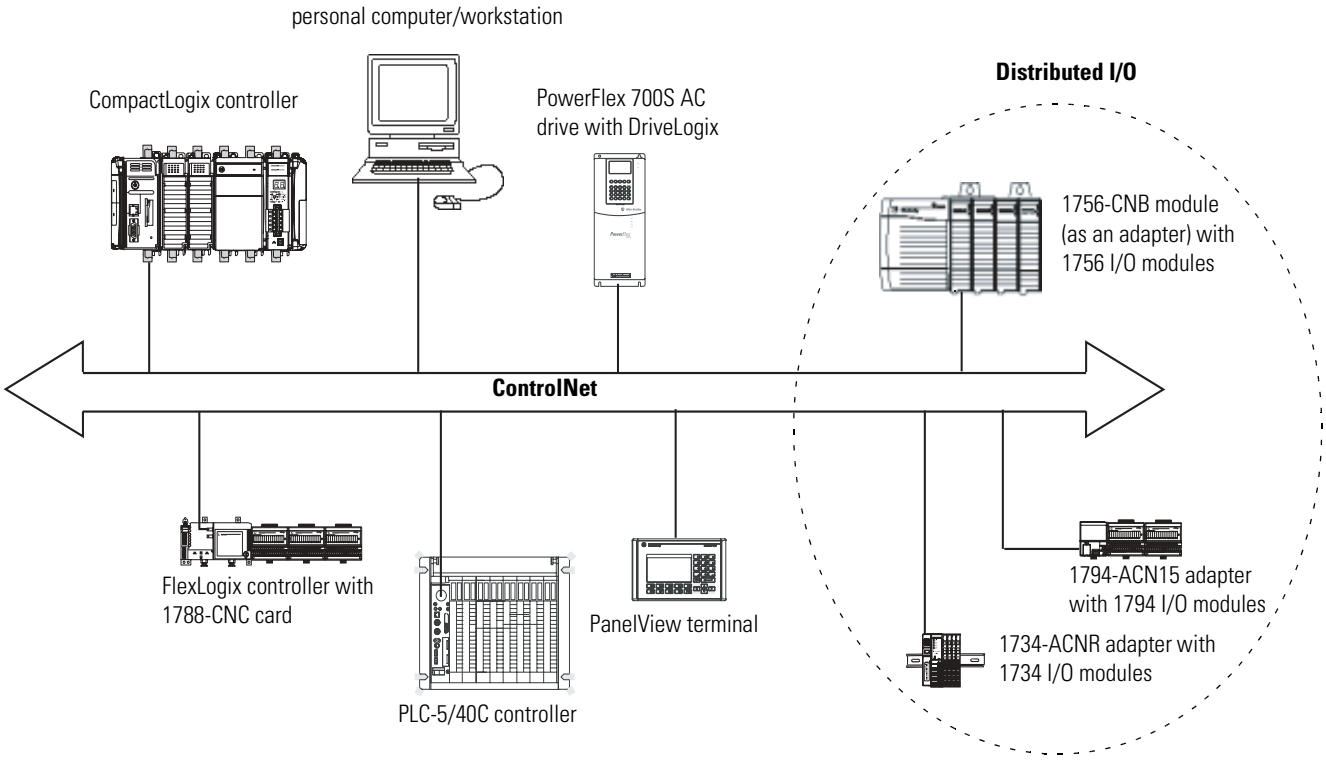
  Use this software to configure the ControlNet network, define the NUT (Network update time), and schedule the ControlNet network.

The ControlNet communications modules:

- support messaging, produced/consumed tags and distributed I/O
- share a common application layer with DeviceNet and EtherNet/IP
- require no routing tables
- support the use of coax and fiber repeaters for isolation and increased distance

In this example:

- The controllers can produce and consume tags among each other.
- The controllers can initiate MSG instructions that send/receive data or configure devices.
- The personal computer can upload/download projects to the controllers.
- The personal computer can configure devices on ControlNet, and it can configure the network itself.



Connections over ControlNet

You indirectly determine the number of connections the controller uses by configuring the controller to communicate with other devices in the system. Connections are allocations of resources that provide more reliable communications between devices compared to unconnected messages.

ControlNet connections can be:

Connection method:	Description:
scheduled (unique to ControlNet)	A scheduled connection is unique to ControlNet communications. A scheduled connection lets you send and receive data repeatedly at a predetermined interval, which is the requested packet interval (RPI). For example, a connection to an I/O module is a scheduled connection because you repeatedly receive data from the module at a specified interval. Other scheduled connections include connections to: • communication devices • produced/consumed tags On a ControlNet network, you must use RSNetWorx for ControlNet to enable all scheduled connections and establish a network update time (NUT). Scheduling a connection reserves network bandwidth to specifically handle the connection.
unscheduled	An unscheduled connection is a message transfer between controllers that is triggered by the requested packet interval (RPI) or the program (such as a MSG instruction). Unscheduled messaging lets you send and receive data when needed. Unscheduled connections use the remainder of network bandwidth after scheduled connections are allocated.

The 1756-CNB, -CNBR communication modules support 64 CIP connections over a ControlNet network. However, it's recommended that you configure only 48 connections for each module to maintain optimal performance.

For more information... The *ControlNet Modules in Logix5000 Control Systems User Manual*, CNET-UM001 provides information on how to:

- configure a ControNet communication module
- control I/O over ControlNet
- send a message over ControlNet
- produce/consume a tag over ControlNet
- calculate controller connections over ControlNet

The *Logix5000 Controllers Design Guidelines Reference Manual*, 1756-RM094 provides guidelines on optimizing a control application on a ControlNet network.

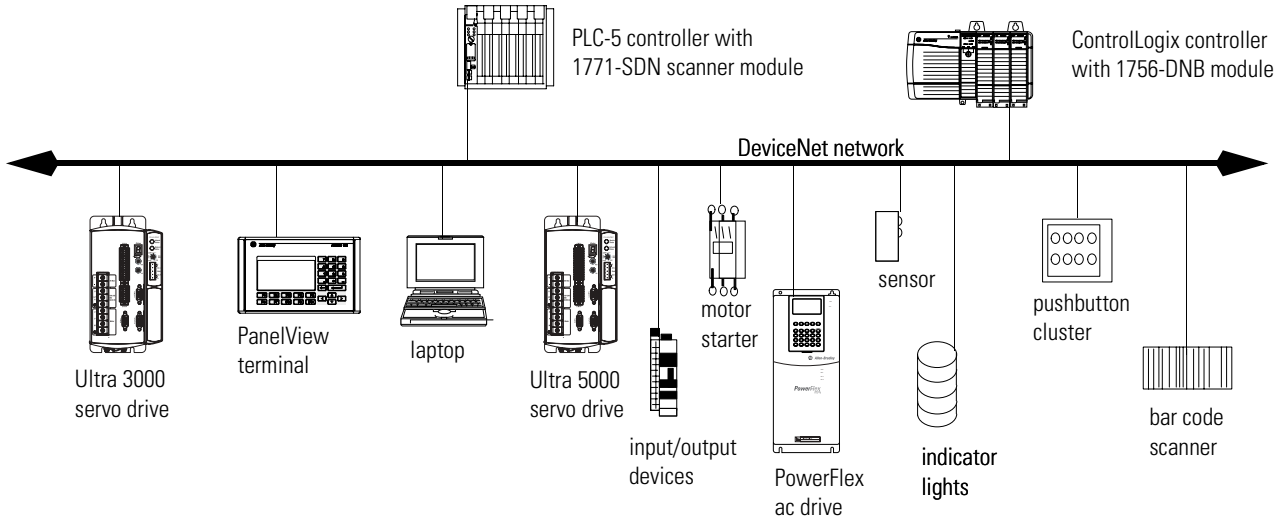
DeviceNet

See:

- *DeviceNet Modules in Logix5000 Control Systems User Manual*, DNET-UM004
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

For DeviceNet communications, use a 1756-DNB module. The DeviceNet network uses the Common Industrial Protocol (CIP) to provide the control, configuration, and data collection capabilities for industrial devices.

If your application:	Select:
• controls I/O modules • requires an adapter for distributed I/O on DeviceNet links • communicates with other DeviceNet devices (messages)	1756-DNB
• links an EtherNet/IP network to a DeviceNet network • has multiple networks	1788-EN2DN
• links a ControlNet network to a DeviceNet network • has multiple networks	1788-CN2DN



In addition to communication hardware for DeviceNet networks, these software products are available:

- RSLogix 5000 programming software is required

  Use this to configure the ControlLogix project and define DeviceNet communications.

- RSNetWorx for DeviceNet is required

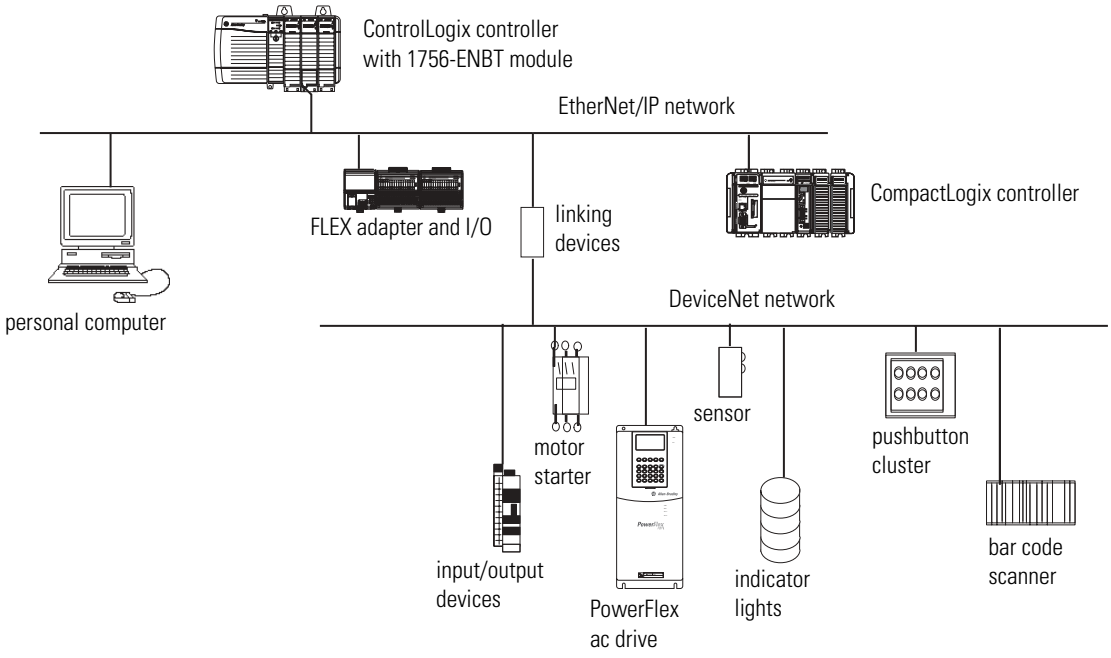
  Use this software to configure DeviceNet devices and define the scan list for those devices.

The DeviceNet communications module:

- supports messaging to devices (not controller to controller)
- shares a common application layer with ControlNet and EtherNet/IP
- offers diagnostics for improved data collection and fault detection
- requires less wiring than traditional, hardwired systems

You can use a linking device as a:

- gateway to connect information- or control-level networks to device-level networks for programming, configuration, control or data collection
- router/bridge to connect the EtherNet/IP or ControlNet network to the DeviceNet network



Connections over DeviceNet

The ControlLogix controller requires two connections for each 1756-DNB module. One connection is for module status and configuration. The other connection is a rack-optimized connection for the device data.

The 1756-DNB module has fixed sections of memory for the input and output data of the DeviceNet devices on the network. Each device on your network requires either some input or output memory of the scanner. Some devices both send and receive data, so they need both input and output memory. The 1756-DNB module supports as many as:

- 124 DINTs of input data
- 123 DINTs of output data

For more information...

The *DeviceNet Modules in Logix5000 Control Systems User Manual*, DNET-UM004 provides information on how to:

- configure a DeviceNet network
- control devices on DeviceNet

The *Logix5000 Controllers Design Guidelines Reference Manual*, 1756-RM094 provides guidelines on optimizing a control application on a DeviceNet network.

Serial

See:

- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001

The ControlLogix controller has one built-in RS-232 port.

If your application:	Select:
• communicates between a controller and other DF1-compatible devices using DF1 protocols • uses modems • controls SCADA applications • controls ASCII devices	built-in serial port
• requires additional RS-232 connections • requires RS-422 and/or RS-485 connections	1756-MVI 1756-MVID

IMPORTANT Limit the length of serial (RS-232) cables to 15.2m (50 ft.).

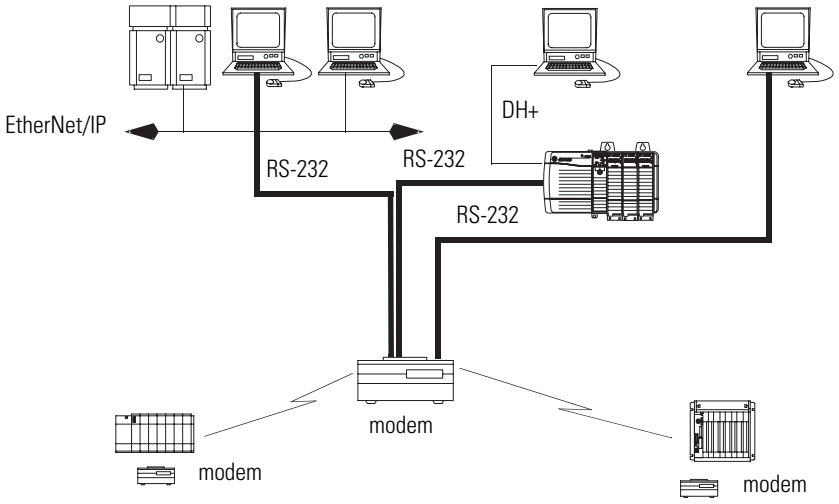
You can configure the serial port of the controller for these modes:

Use this mode:	For:
DF1 point-to-point	communication between the controller and one other DF1-protocol-compatible device. This is the default system mode. Default parameters are: • Baud Rate: 19200 • Data Bits: 8 • Parity: None • Stop Bits: 1 • Control Line: No Handshake • RTS send Delay: 0 • RTS Off Delay: 0 This mode is typically used to program the controller through its serial port.
DF1 master mode	control of polling and message transmission between the master and slave nodes. The master/slave network includes one controller configured as the master node and as many as 254 slave nodes. Link slave nodes using modems or line drivers. A master/slave network can have node numbers from 0 to 254. Each node must have a unique node address. Also, at least 2 nodes must exist to define your link as a network (1 master and 1 slave station are the two nodes).
DF1 slave mode	using a controller as a slave station in a master/slave serial communication network. When there are multiple slave stations on the network, link slave stations using modems or line drivers to the master. When you have a single slave station on the network, you do not need a modem to connect the slave station to the master. You can configure the control parameters for no handshaking. You can connect 2 to 255 nodes to a single link. In DF1 slave mode, a controller uses DF1 half-duplex protocol. One node is designated as the master and it controls who has access to the link. All the other nodes are slave stations and must wait for permission from the master before transmitting.
User mode	communicating with ASCII devices. This requires your program to use ASCII instructions to read and write data from and to an ASCII device.
DH-485	communicating with other DH-485 devices multi-master, token passing network allowing programming and peer-to-peer messaging.

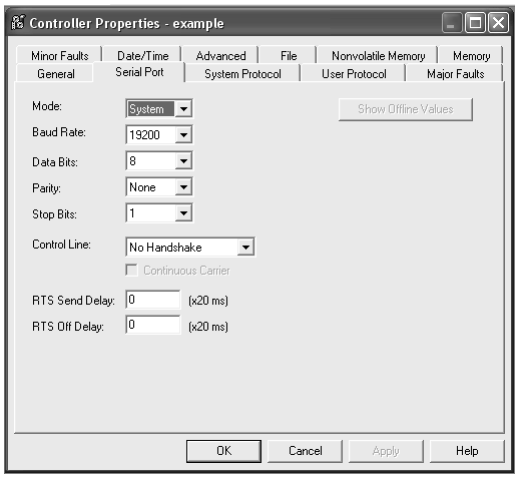
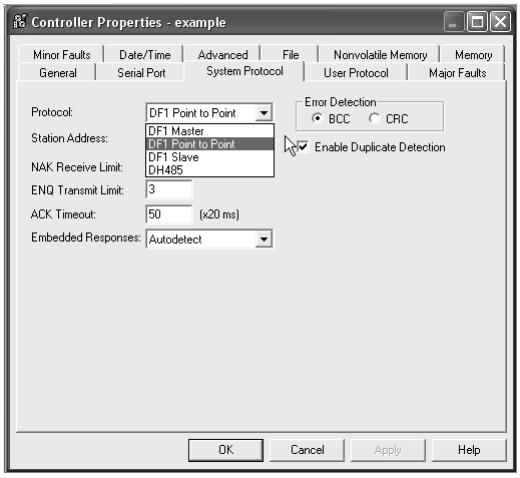
Communicate with DF1 devices

You can configure the controller as a master or slave on a serial communication network. Use a serial to network get information to and from remote controllers (stations) when:

- the system contains three or more stations
- communications occur on a regular basis and require leased-line, radio, or power-line modems



To configure the controller for DF1 communications:

On this tab:	Do this:
	1. Select System Mode 2. Specify communication settings
	1. Select DF1 protocol 2. Specify DF1 settings

For more information... The *Logix5000 Controllers General Instructions Reference Manual*, 1756-RM003 defines the instructions you can use to manipulate ASCII characters.

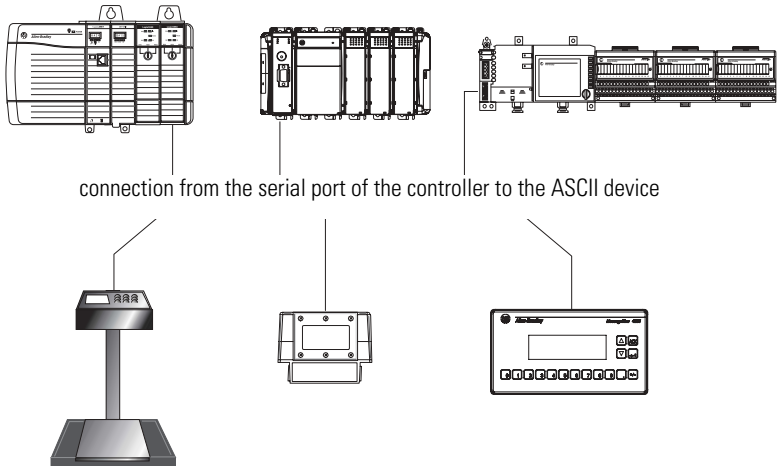
The *SCADA System Application Guide*, AG-UM008 provides information on how to:

- select a polling mode
- configure controllers, modems, and software
- troubleshoot basic DF1 protocol issues

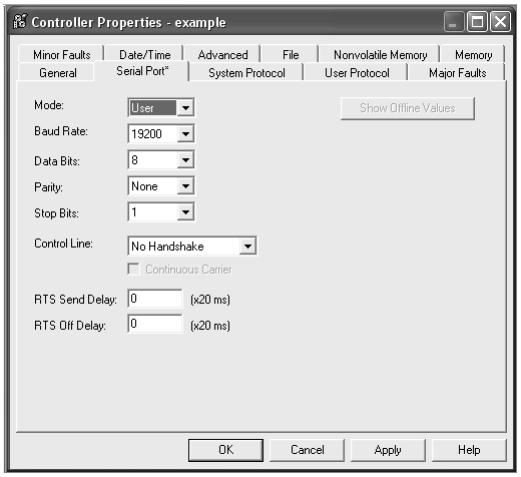
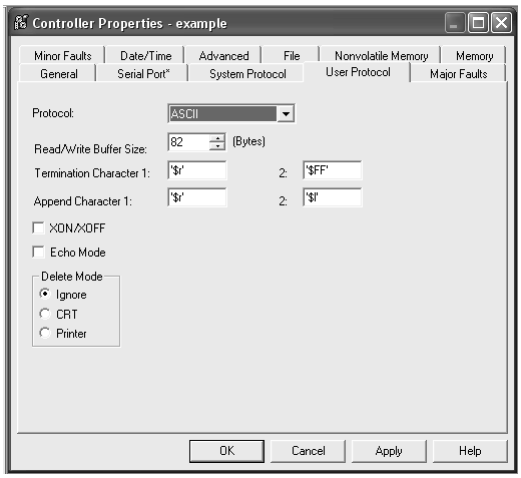
Communicate with ASCII devices

When configured for user mode, you can use the serial port to interface with ASCII devices. For example, you can use the serial port to:

- read ASCII characters from a weigh scale module or bar code reader
- send and receive messages from an ASCII triggered device, such as a MessageView terminal.



To configure the controller for DF1 communications:

On this tab:	Do this:
	1. Select User Mode 2. Specify communication settings
	1. Select ASCII protocol 2. Specify ASCII character settings

The controller supports several instructions to manipulate ASCII characters. The instructions are available in ladder diagram (LD) and structured text (ST).

Read and write ASCII characters

If you want to:	Use this instruction:
determine when the buffer contains termination characters	ABL
count the characters in the buffer	ACB
clear the buffer	ACL
clear out ASCII Serial Port instructions that are currently executing or are in the queue	
obtain the status of the serial port control lines	AHL
turn on or off the DTR signal	
turn on or off the RTS signal	
read a fixed number of characters	ARD
read a varying number of characters, up to and including the first set of termination characters	ARL
send characters and automatically append one or two additional characters to mark the end of the data	AWA
send characters	AWT

Create and modify strings of ASCII characters

If you want to:	Use this instruction:
add characters to the end of a string	CONCAT
delete characters from a string	DELETE
determine the starting character of a sub-string	FIND
insert characters into a string	INSERT
extract characters from a string	MID

Convert data to or from ASCII characters

If you want to:	Use this instruction:
convert the ASCII representation of an integer value to a SINT, INT, DINT, or REAL value	STOD
convert the ASCII representation of a floating-point value to a REAL value	STOR
convert a SINT, INT, DINT, or REAL value to a string of ASCII characters	DTOS
convert a REAL value to a string of ASCII characters	RTOS
convert the letters in a string of ASCII characters to upper case	UPPER
convert the letters in a string of ASCII characters to lower case	LOWER

For more information... The *Logix5000 Controllers General Instructions Reference Manual*, 1756-RM003 defines the instructions you can use to manipulate ASCII characters.

The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- communicate with an ASCII device
- transmit/receive ASCII characters

Modbus support

See:
- *Logix5000 Controllers as Masters or Slaves on Modbus Application Solution*, CIG-AP129

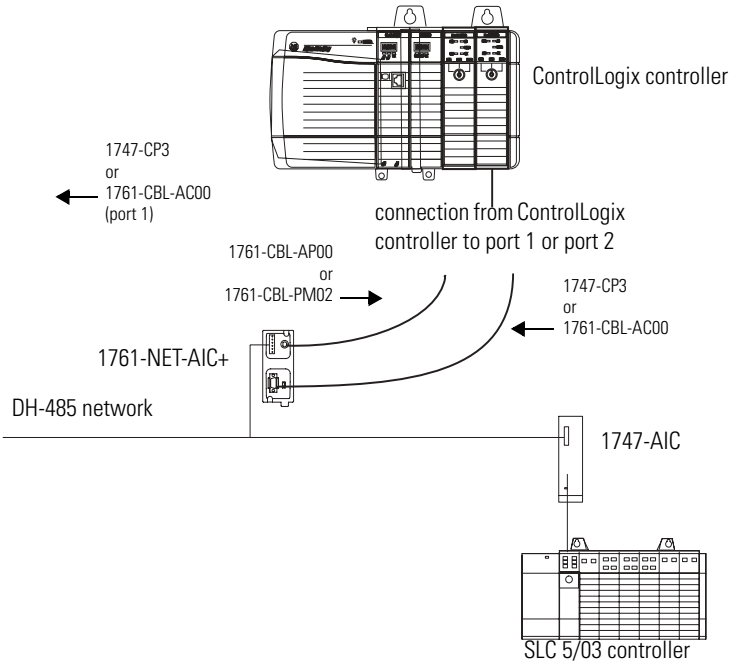
To use Logix5000 controllers on Modbus, you connect through the serial port and execute specific ladder logic routines. A sample controller project is available with RSLogix 5000 Enterprise programming software. From RSlogix 5000 software, select Help → Vendor Sample Projects to display a list of available, sample projects.

DH-485

For DH-485 communication, use the serial port of the controller. However, when using a ControlLogix controller, it is recommended that you use NetLinx networks (EtherNet/IP, ControlNet, or DeviceNet) because excessive traffic on a DH-485 network may make it impractical to connect to a controller with RSLogix 5000 programming software.

If your application uses:	Select:
• connections to existing DH-485 networks	built-in serial port

The DH-485 protocol uses RS-485 half-duplex as its physical interface. (RS-485 is a definition of electrical characteristics; it is not a protocol.) You can configure the RS-232 port of the ControlLogix controller to act as a DH-485 interface. By using a 1761-NET-AIC and the appropriate RS-232 cable (1756-CP3 or 1747-CP3), a ControlLogix controller can send and receive data on a DH-485 network.



On the DH-485 network, the ControlLogix controller can send and receive messages to and from other controllers on the network

For the controller to operate on a DH-485 network, you need:

- a 1761-NET-AIC interface converter for each controller you want to put on the DH-485 network.

  You can have two controllers per one 1761-NET-AIC converter, but you need a different cable for each controller. Connect the serial port of the controller to either port 1 or port 2 of the 1761-NET-AIC converter. Use the RS-485 port to connect the converter to the DH-485 network.

  The cable you use to connect the controller depends on the port you use on the 1761-NET-AIC converter.

If you connect to this port:	Use this cable:
port 1 DB-9 RS-232, DTE connection	1747-CP3 or 1761-CBL-AC00
port 2 mini-DIN 8 RS-232 connection	1761-CBL-AP00 or 1761-CBL-PM02

- RSLogix 5000 programming software to configure the serial port of the controller for DH-485 communications.

  Specify these characteristics on the Serial Port tab (default values are shown in bold):

Characteristic:	Description:
Baud Rate	Specifies the communication rate for the DH-485 port. All devices on the same DH-485 network must be configured for the same baud rate. Select 9600 or 19200 Kbps.
Node Address	Specifies the node address of the controller on the DH-485 network. Select a number 1-31 decimal, inclusive. To optimize network performance, assign node addresses in sequential order. Initiators, such as personal computers, should be assigned the lowest address numbers to minimize the time required to initialize the network.
Token Hold Factor	Number of transmissions (plus retries) that a node holding a token can send onto the data link each time that it receives the token. Enter a value between 1-4. The default is 1.
Maximum Node Address	Specifies the maximum node address of all the devices on the DH-485 network. Select a number 1-31 decimal, inclusive. To optimize network performance, make sure: • the maximum node address is the highest node number being used on the network • that all the devices on the same DH-485 network have the same selection for the maximum node address.

DH+

See:

- *ControlLogix Data Highway Plus and Remote I/O Communication Interface Module User Manual*, 1756-UM514

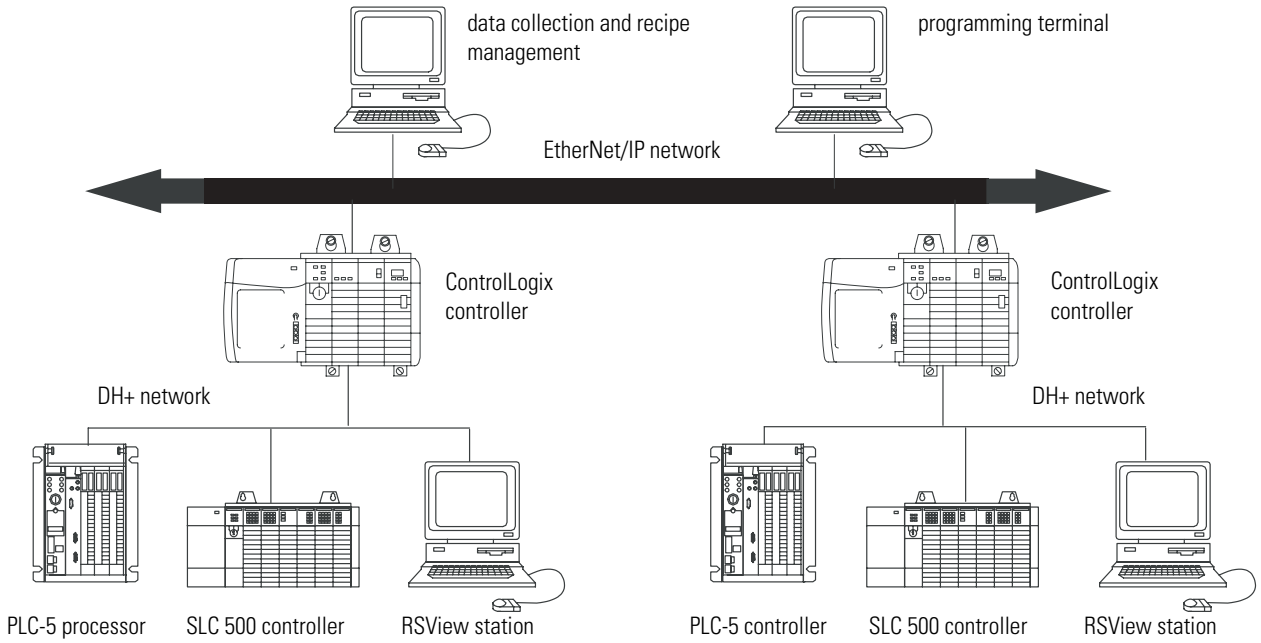
For DH+ communications, use a 1756-DHRIO module to exchange information between:

- PLC controllers and SLC controllers
- ControlLogix controllers and PLC or SLC controllers
- ControlLogix controllers

If your application uses:	Select:
• plantwide and cell-level data sharing with program maintenance • data sent regularly • transfer of information between controllers	1756-DHRIO

You can connect a maximum of 32 stations to a single DH+ link. Channel A supports 57.6 Kbps, 115.2 Kbps, and 230.4 Kbps. Channel B supports 57.6 Kbps and 115.2 Kbps.

In this example, the two ControlLogix chassis link existing DH+ networks. The PLC-5 and SLC controllers can communicate with the devices on their own DH+ network as well as the devices on the other DH+ network.



Communicate over DH+

For the controller to communicate to a workstation or other device over a DH+ network, use RSLinx software to:

- specify a unique link ID for each ControlLogix backplane and additional network in the communication path
- configure the routing table for the 1756-DHRIO module

The 1756-DHRIO module can route a message through as many as four communications networks and three chassis. This limit applies only to the routing of a message and not to the total number of networks or chassis in a system.

For more information... The *ControlLogix Data Highway Plus - Universal Remote I/O Module User Manual*, 1756-UM514 provides information on how to:

- configure the module for DH+ communications
- send a message over DH+

Universal Remote I/O

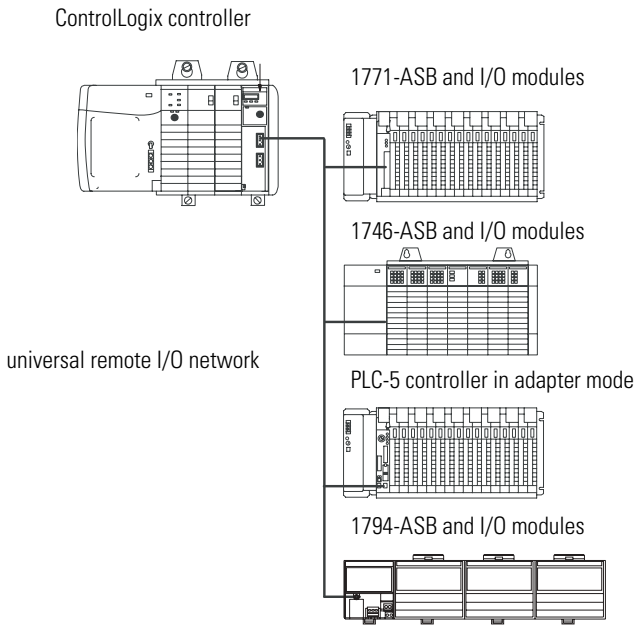
For universal remote I/O communications, use a 1756-DHRIO module.

See:

- *ControlLogix Data Highway Plus and Remote I/O Communication Interface Module User Manual*, 1756-UM514
- *Process Remote I/O Interface Module User Manual*, 1757-UM007

If your application uses:	Select:
• connections between controllers and I/O adapters • data sent regularly • distributed control so that each controller has its own I/O and • communicates with a supervisory controller	1756-DHRIO
• needs an RIO scanner • communicates with as many as 32 RIO adapters • supports HART (Highway Addressable Remote Transducer) devices • requires scheduled connections to update data on a ControlLogix controller	1757-ABRIO

When a channel on the 1756-DHRIO module is configured for remote I/O, the module acts as a scanner for a universal remote I/O network. The controller communicates to the module to send and receive the I/O on the universal remote I/O network.



Communicate over universal remote I/O

For the controller to control I/O over a universal remote I/O network, you must:

1. Configure the remote I/O adapter.
2. Layout the remote I/O network cable.
3. Connect the remote I/O network cable.
4. Configure the scanner channel.

Keep these rules in mind as you design a remote I/O network:

- All devices connected to a remote I/O network must communicate using the same communication rate. The following rates are available for remote I/O:
  - 57.6 Kbps
  - 115.2 Kbps
  - 230.4 Kbps
- Assign unique partial and full racks to each channel used in remote I/O scanner mode. Both channels of a 1756-DHRIO module cannot scan the same partial or full rack address. Both module channels can communicate to 00-37 octal or 40-77 octal, but each channel can only communicate with address in one of the ranges at a time.
- A channel can have a maximum of 32 rack numbers and a maximum of 32 physical devices connected to it.
- A channel can have a maximum of 16 block-transfer connections.

For more information...

The *ControlLogix Data Highway Plus - Universal Remote I/O Module User Manual*, 1756-UM514 provides information on how to:

- configure the module for universal remote I/O communications
- control I/O over universal remote I/O

The *Process Remote I/O Interface Module User Manual*, 1757-UM007 provides information on how to:

- configure the module for universal remote I/O communications
- control I/O over universal remote I/O

FOUNDATION Fieldbus

See:

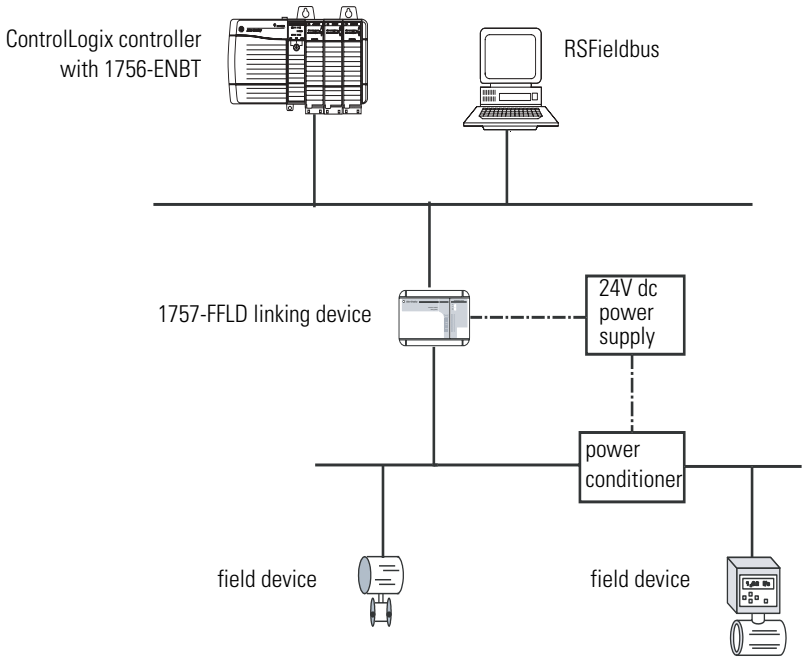
- *RSFieldbus User Manual*, RSFBUS-UM001
- *RSFieldbus Application Guide*, RSFBUS-AT001

FOUNDATION Fieldbus is an open interoperable fieldbus designed for process control instrumentation. For FOUNDATION Fieldbus, select from these devices:

If your application uses:	Select:
• bridges EtherNet/IP to FOUNDATION Fieldbus • low-speed serial (H1) and high-speed Ethernet (HSE) connections • OPC server for direct access to devices	1757-FFLD
• low-speed serial (H1) connections • bridges ControlNet to FOUNDATION Fieldbus • redundant ControlNet media	1788-CN2FF

FOUNDATION Fieldbus allows control to be distributed to and executed in the device. The FOUNDATION Fieldbus module:

- bridges from Ethernet to H1
- accepts either HSE or EtherNet/IP messages and converts them to the H1 protocol



HART (Highway Addressable Remote Transducer) Protocol)

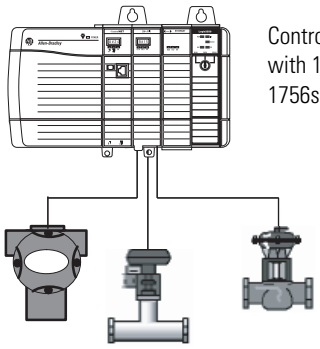
See:

- *FLEX Ex HART Analog Modules User Manual*, 1797-6.5.3
- Encompass web site at www.automation/rockwell/encompass

HART is an open protocol designed for process control instrumentation. For HART connectivity, select from these devices:

If your application uses:	Select:
• data acquisition or control application with slow update requirements (such as a tank farm) • no external hardware required to access HART signal • does not connect directly to asset management software	Prosoft interface MVI56-HART
• analog and HART in one module • no external hardware required to access HART signal • HART commands can be transmitted as unscheduled messages • supports asset management software to HART device	Spectrum analog I/O modules 1756sc-IF8H 1756sc-OF8H
• analog and HART in one module • instrumentation in hazardous locations (FLEX Ex) • HART commands can be transmitted as unscheduled messages • directly connects asset management software to HART devices	1794 FLEX I/O • 1794-IE8H • 1794-OE8H 1797 FLEX Ex I/O • 1797-IE8H • 1797-OE8H

The HART protocol combines digital signals with analog signals so the digital signal can be used for the process variable (PV). The HART protocol also provides diagnostic data from the transmitter.



Notes:

Manage Controller Communications

Use This Chapter

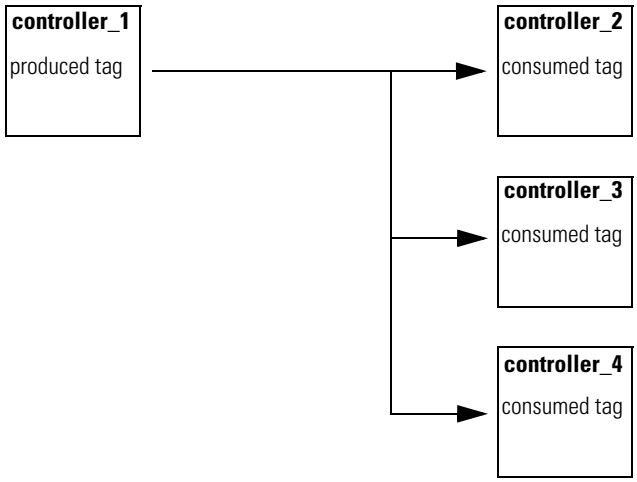
For this information:	See:
Produce and Consume (Interlock) Data	4-1
Send and Receive Messages	4-3
Connection Overview	4-4
Calculate Connection Use	4-5
Connections Example	4-6

Produce and Consume (Interlock) Data

See:

- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

The controller supports the ability to produce (broadcast) and consume (receive) system-shared tags over ControlNet or EtherNet/IP networks. Produced and consumed tags each require connections. Over ControlNet, produced and consumed tags are scheduled connections.



This type of tag:	Description:
produced	A produced tag allows other controllers to consume the tag, which means that a controller can receive the tag data from another controller. The producing controller uses one connection for the produced tag and one connection for each consumer. The controller's communication device uses one connection for each consumer. As you increase the number of controllers that can consume a produced tag, you also reduce the number of connections the controller and communication device have available for other operations, like communications and I/O.
consumed	Each consumed tag requires one connection for the controller that is consuming the tag. The controller's communication device uses one connection for each consumer.

For two controllers to share produced or consumed tags, both controllers must be attached to the same control network (such as a ControlNet or Ethernet/IP network). You cannot bridge produced and consumed tags over two networks.

The total number of tags that can be produced or consumed is limited by the number of available connections. If the controller uses all of its connections for I/O and communication devices, no connections are left for produced and consumed tags.

For more information...

The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- produce a tag
- consume a tag
- produce a large array

The *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094 provides guidelines on how to:

- create produced and consumed tags
- specify an RPI
- manage connections

Send and Receive Messages

See:

- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

Messages transfer data to other devices, such as other controllers or operator interfaces. Messages use unscheduled connections to send or receive data. Connected messages can leave the connection open (cache) or close the connection when the message is done transmitting.

Connected messages are unscheduled connections on both ControlNet and EtherNet/IP networks. Each message uses one connection, regardless of how many devices are in the message path.

This message type:	With this communication method:	Is a connected message:	Which you can cache
CIP data table read or write	CIP	✓	✓
PLC2, PLC3, PLC5, or SLC (all types)	CIP	✓	✓
	CIP with Source ID	✓	✓
	DH+	✓	
CIP generic		your option (1)	✓
block-transfer read or write	na	✓	✓

(1) You can connect CIP generic messages. But for most applications we recommend you leave CIP generic messages unconnected.

Determine whether to cache message connections

When you configure a MSG instruction, you have the option of whether or not to cache the connection.

If the message executes:	Then:
repeatedly	Cache the connection. This keeps the connection open and optimizes execution time. Opening a connection each time the message executes increases execution time.
infrequently	Do not cache the connection. This closes the connection upon completion of the message, which frees up that connection for other uses.

For more information... The *Logix5000 Controllers General Instructions Reference Manual*, 1756-RM003 describes how to use the MSG instruction.

The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- execute a MSG instruction
- get and set the number of unconnected buffers
- convert INT data to DINT data
- manage multiple MSG instructions
- send one MSG to multiple devices

Connection Overview

See:

- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

A Logix5000 system uses a connection to establish a communication link between two devices. Connections can be:

- controller to local I/O modules or local communication modules
- controller to remote I/O or remote communication modules
- controller to remote I/O (rack-optimized) modules
- produced and consumed tags
- messages
- controller access by RSLogix 5000 programming software
- controller access by RSLinx software for HMI or other applications

The limit of connections may ultimately reside in the communication module you use for the connection. If a message path routes through a communication module, the connection related to the message also counts towards the connection limit of that communication module.

This device:	Supports this many connections:
ControlLogix controller	250
1756-ENBT 1756-EWEB	128
1756-CNB 1756-CNBR	64 (48 recommended maximum)

Other controllers and communication modules support different maximum numbers of connections.

For more information... The *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094 describes how to optimize connection use.

Calculate Connection Use

To calculate the total number of local connections the controller uses:

Local Connection Type:	Device Quantity:	Connections per Device:	Total Connections:
local I/O module (always a direct connection)		1	
1756-M16SE, -M08SE, -M02AE servo module		3	
1756-CNB, -CNBR ControlNet communication module		0	
1756-ENBT EtherNet/IP communication module		0	
1756-EWEB EtherNet/IP web server module		0	
1756-DNET DeviceNet communication module		2	
1756-DHRIO DH+/Universal remote I/O communication module		1	
		total	

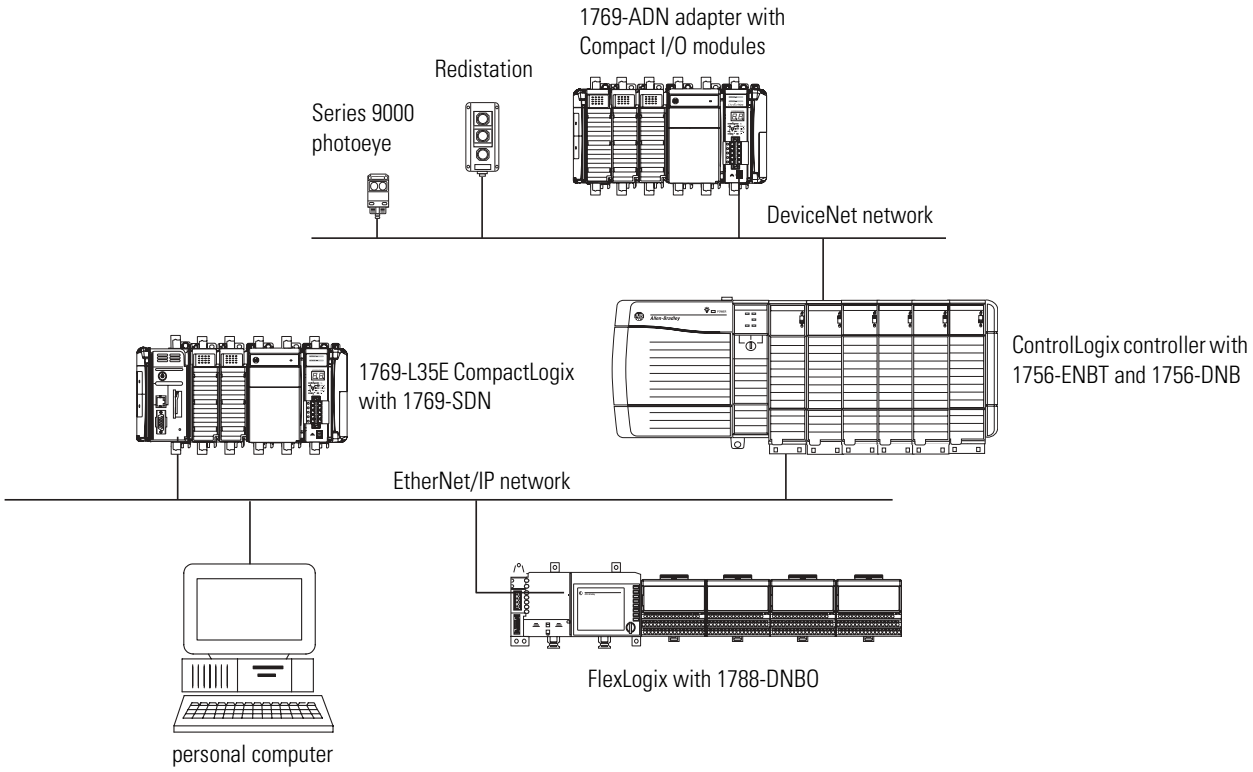
Remote connections depend on the communication module. The number of connections the module itself supports determines how many connections the controller can access through that module. To calculate the total number of remote connections the controller uses:

Remote Connection Type:	Device Quantity:	Connections per Device:	Total Connections:
remote ControlNet communication module I/O configured as direct connection (none) I/O configured as rack-optimized connection		 0 **or** 1	
remote I/O module over ControlNet (direct connection)		1	
remote EtherNet/IP communication module I/O configured as direct connection (none) I/O configured as rack-optimized connection		 0 **or** 1	
remote I/O module over EtherNet/IP (direct connection)		1	
remote device over DeviceNet (accounted for in rack-optimized connection for local 1756-DNB)		0	
other remote communication adapter		1	
produced tag each consumer		1 1	
consumed tag		1	
message (depending on type)		1	
block-transfer message		1	
		total	

Connections Example

In this example system the 1756 ControlLogix controller:

- controls local (in the same chassis) digital I/O modules
- controls remote I/O devices on DeviceNet
- sends and receives messages to/from a CompactLogix controller on EtherNet/IP
- produces one tag that the 1794 FlexLogix controller consumes
- is programmed via RSLogix 5000 programming software



The ControlLogix controller in this example uses these connections:

Connection Type:	Device Quantity:	Connections per Device:	Total Connections:
controller to local I/O modules	4	1	1
controller to 1756-ENBT module	1	0	0
controller to 1756-DNB module	1	2	2
controller to RSLogix 5000 programming software	1	1	1
message to CompactLogix controller	2	1	2
produced tag consumed by FlexLogix controller	1 1	1 1	1 1
		total	8

Place, Configure, and Monitor I/O

Use This Chapter

For this information:	See:
Select I/O Modules	5-1
Place Local I/O Modules	5-2
Configure I/O	5-2
Configure Distributed I/O on EtherNet/IP	5-5
Configure Distributed I/O on ControlNet	5-6
Configure Distributed I/O on DeviceNet	5-7
Address I/O Data	5-8
Add 1756 I/O at Runtime	5-8
Determine When Data Is Updated	5-11
Reconfigure an I/O Module	5-12

Select I/O Modules

See:

- *ControlLogix Selection Guide,* 1756-SG001

When selecting 1756 I/O modules, select:

- specialty I/O modules, when appropriate - some modules have field-side diagnostics, electronic fusing, or individually-isolated inputs/outputs
- 1756 remote terminal blocks (RTBs) or 1492 wiring systems for the I/O modules
- 1492 PanelConnect modules and cables to connect input modules to sensors

Place Local I/O Modules

See:
- *ControlLogix Digital I/O Modules User Manual*, 1756-UM058
- *ControlLogix Analog I/O Modules User Manual*, 1756-UM009

The 1756 chassis you use determines how many local I/O modules you can use. The slots in a chassis can be filled with any combination of controllers, communication modules, and I/O modules.

This chassis:	Has this many available slots:
1756-A4	4
1756-A7	7
1756-A10	10
1756-A13	13
1756-A17	17

For empty slots, use the slot-filler module, 1756-N2.

The ControlLogix controller also supports distributed (remote) I/O via these networks:

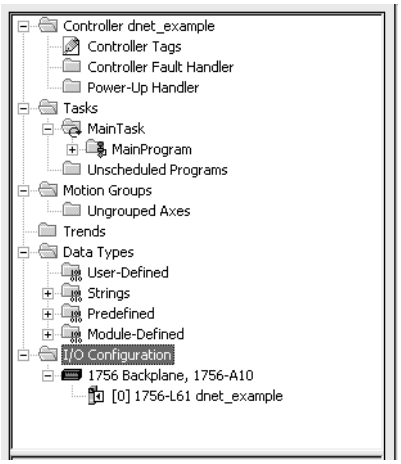
- EtherNet/IP
- ControlNet
- DeviceNet
- Universal remote I/O

Configure I/O

See:
- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

To communicate with an I/O module in your system, you add the module to the I/O Configuration folder of the controller.

Add I/O modules to the 1756 backplane.



When you add a module, you also define a specific configuration for the module. While the configuration options vary from module to module, there are some common options that you typically configure:

Configuration Option:	Description:
requested packet interval (RPI)	The RPI specifies the period at which data updates over a connection. For example, an input module sends data to a controller at the RPI that you assign to the module. • Typically, you configure an RPI in milliseconds (ms). The range is 0.2 ms (200 microseconds) to 750 ms. • If a ControlNet network connects the devices, the RPI reserves a slot in the stream of data flowing across the ControlNet network. The timing of this slot may not coincide with the exact value of the RPI, but the control system guarantees that the data transfers at least as often as the RPI.
change of state (COS)	Digital I/O modules use change of state (COS) to determine when to send data to the controller. If a COS does not occur within the RPI timeframe, the module multicasts data at the rate specified by the RPI. Because the RPI and COS functions are asynchronous to the logic scan, it is possible for an input to change state during program scan execution. If this is a concern, buffer input data so your logic has a stable copy of data during its scan. Use the Synchronous Copy (CPS) instruction to copy the input data from your input tags to another structure and use the data from that structure.
communication format	Many I/O modules support different formats. The communication format that you choose also determines: • data structure of tags • connections • network usage • ownership • whether the module returns diagnostic information
electronic keying	When you configure a module, you specify the slot number for the module. However, it is possible to place a different module in that slot, either on purpose or accidently. Electronic keying lets you protect your system against the accidental placement of the wrong module in a slot. The keying option you choose determines how closely any module in a slot must match the configuration for that slot before the controller opens a connection to the module. There are different keying options depending on your application needs.

I/O connections

A Logix5000 system uses connections to transmit I/O data. A connection can be:

Connection:	Description:
direct	A direct connection is a real-time, data transfer link between the controller and an I/O module. The controller maintains and monitors the connection between the controller and the I/O module. Any break in the connection, such as a module fault or the removal of a module while under power, causes the controller to set fault status bits in the data area associated with the module. Typically, analog I/O modules, diagnostic I/O modules, and specialty modules require direct connections.
rack-optimized	For digital I/O modules, you can select rack-optimized communication. A rack-optimized connection consolidates connection usage between the controller and all the digital I/O modules on a rack (or DIN rail). Rather than having individual, direct connections for each I/O module, there is one connection for the entire rack (or DIN rail).

For more information... The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- configure I/O
- address I/O data
- buffer I/O data

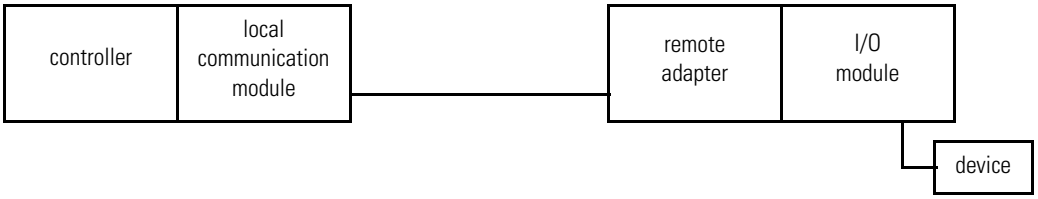
The *Logix5000 Controllers Design Guidelines Reference Manual*, 1756-RM094 provides guidelines on how to:

- buffer I/O
- specify an RPI rate
- select a communication format
- manage I/O connections

Configure Distributed I/O on EtherNet/IP

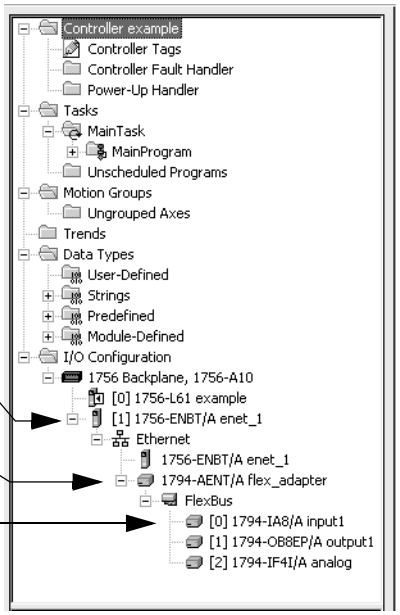
To communicate with the I/O modules over EtherNet/IP, you add the EtherNet/IP bridge, EtherNet/IP adapter, and I/O modules to the I/O Configuration folder of the controller. Within the I/O Configuration folder, you organize the modules into a hierarchy (tree/branch, parent/child).

For a typical distributed I/O network…



…you build the I/O configuration in this order

1. Add the local communication module (bridge).
2. Add the remote adapter for the distributed I/O chassis or DIN rail.
3. Add the I/O module.

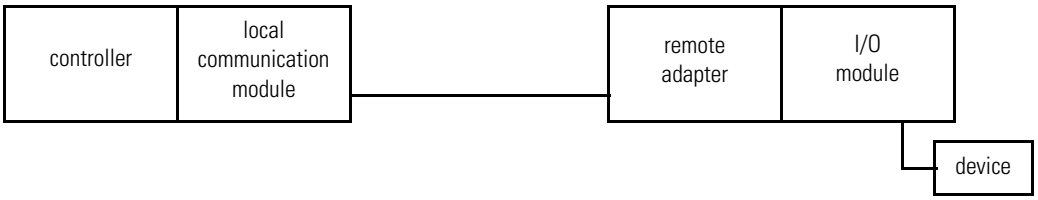


For more information... See *EtherNet/IP Communication Modules in Logix5000 Control Systems User Manual,* ENET-UM001.

Configure Distributed I/O on ControlNet

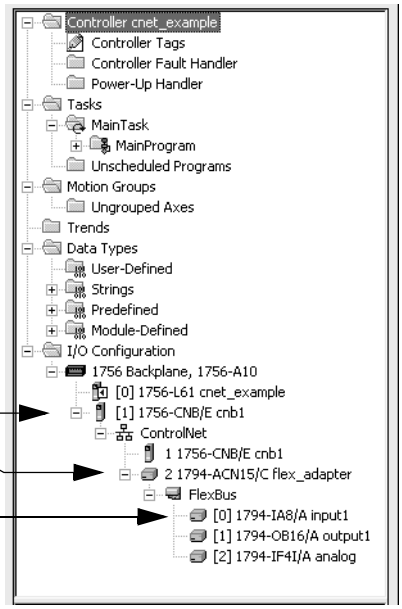
To communicate with the I/O modules over ControlNet, you add the ControlNet bridge, ControlNet adapter, and I/O modules to the I/O Configuration folder of the controller. Within the I/O Configuration folder, you organize the modules into a hierarchy (tree/branch, parent/child).

For a typical distributed I/O network…



…you build the I/O configuration in this order

1. Add the local communication module (bridge).
2. Add the remote adapter for the distributed I/O chassis or DIN rail.
3. Add the I/O module.



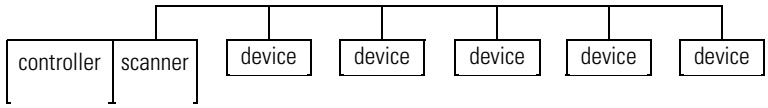
For more information... See *ControlNet Communication Modules in Logix5000 Control Systems User Manual,* CNET-UM001.

Configure Distributed I/O on DeviceNet

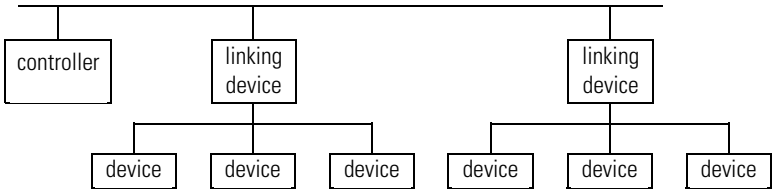
To communicate with the I/O modules over Device, you add the DeviceNet bridge to the I/O Configuration folder of the controller. You define a scanlist within the DeviceNet adapter to communicate data between devices and the controller.

For a typical distributed I/O network…

single network

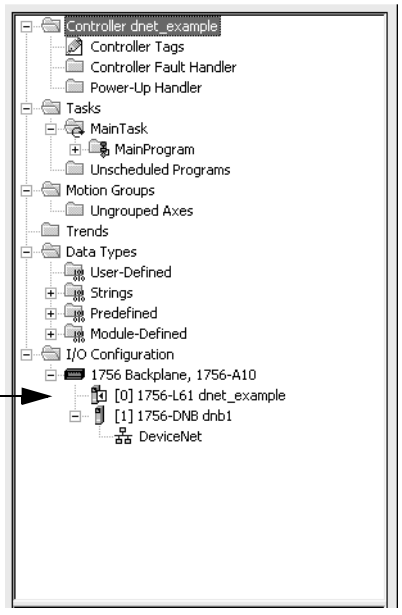


several smaller distributed networks (subnets)



…you build the I/O configuration in this order

Add the local communication module (bridge).



For more information... See *DeviceNet Communication Modules in Logix5000 Control Systems User Manual,* DNET-UM004.

Address I/O Data

I/O information is presented as a set of tags.

- Each tag uses a structure of data. The structure depends on the specific features of the I/O module.
- The name of the tags is based on the location of the I/O module in the system.

An I/O address follows this format:

`Location` `:Slot` `:Type` `.Member` `.SubMember` `.Bit`

= Optional

Where:	Is:
Location	Network location LOCAL = same chassis or DIN rail as the controller *ADAPTER_NAME* = identifies remote communication adapter or bridge module
Slot	Slot number of I/O module in its chassis or DIN rail
Type	Type of data I = input O = output C = configuration S = status
Member	Specific data from the I/O module; depends on what type of data the module can store. • For a digital module, a Data member usually stores the input or output bit values. • For an analog module, a Channel member (CH#) usually stores the data for a channel.
SubMember	Specific data related to a Member.
Bit	Specific point on a digital I/O module; depends on the size of the I/O module (0-31 for a 32-point module)

Add 1756 I/O at Runtime

With RSLogix 5000 programming software, version 15, you can add 1756 I/O modules to the Controller Organizer at runtime:

- You can only add 1756 I/O modules at runtime
- You can add the 1756 I/O modules to the local chassis, remotely via the unscheduled portion of a ControlNet network, and remotely via an EtherNet/IP network.

Considerations when adding ControlNet I/O at runtime

- The ControlNet I/O modules you add at runtime use can be added to existing rack-optimized connections or added as direct connections (you cannot create new rack-optimized connections when adding ControlNet I/O modules at runtime).
- Disable the Change of State (COS) feature on digital input modules because it can cause inputs to be sent faster than the RPI.
- Dedicate one ControlNet network to I/O communications only. On the dedicated I/O network, make sure there is:
  - no HMI traffic
  - no MSG traffic
  - no programming workstations
- RPIs faster than 25 msec for unscheduled modules will overload the 1756-CNB, -CNBR communication module. Also:
  - use a NUT if 10 msec or more
  - keep the SMAX and UMAX values as small as possible
- You can add I/O modules until you reach:
  - 75% utilization of the 1756-CNB, -CNBR communication module
  - depending on the RPI, utilization increase 1-4% for each I/O module you add
  - 48 connections on the 1756-CNB, -CNBR communication module
  - < 350,000 bytes as the remaining unscheduled bandwidth on the ControlNet network

Considerations when adding EtherNet/IP I/O at runtime

Adding EtherNet/IP I/O at runtime follows the same guidelines as adding EtherNet/IP I/O when offline. EtherNet/IP I/O is always unscheduled.

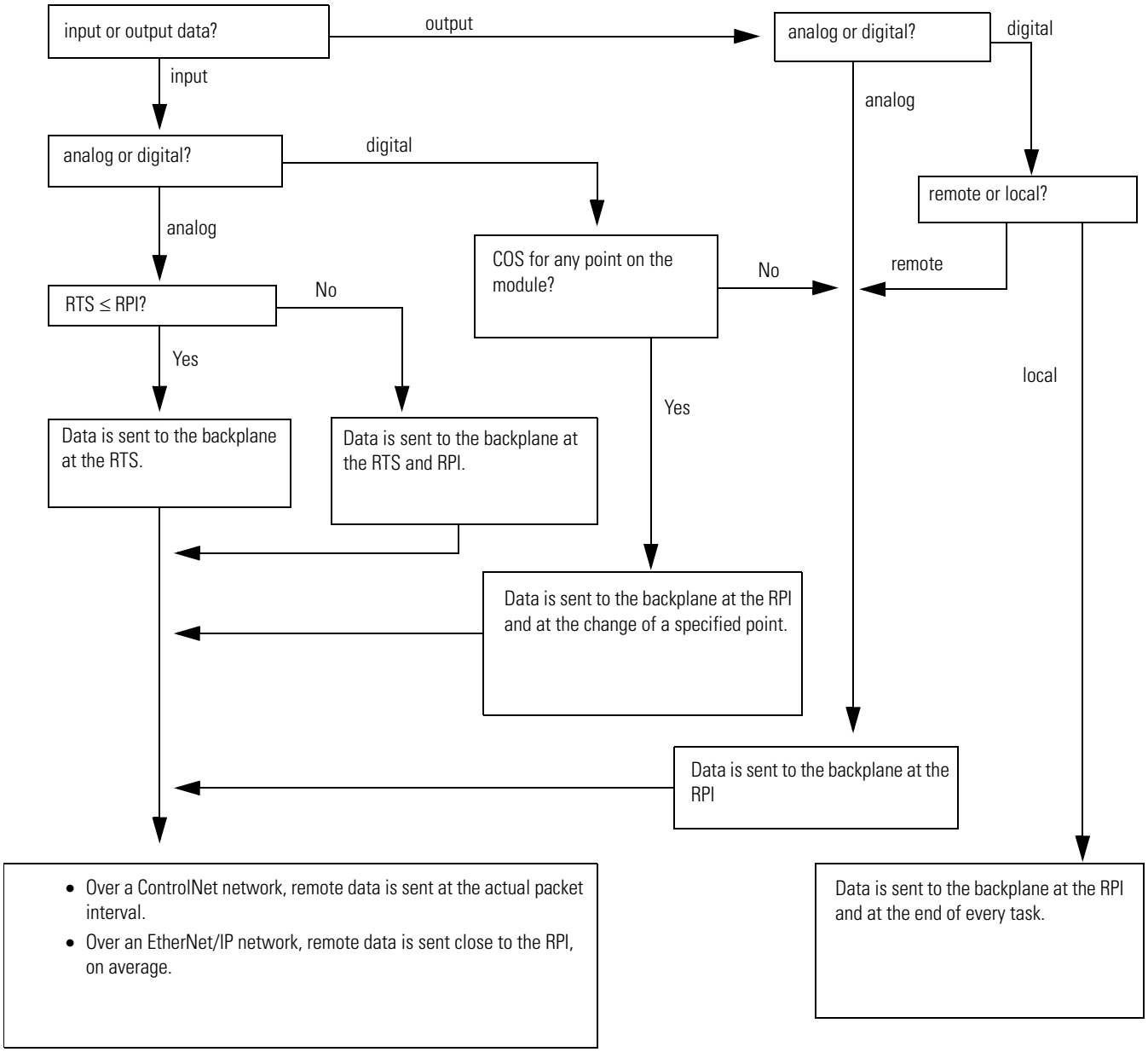
- The EtherNet/IP I/O modules you add at runtime use can be added to existing rack-optimized connections, added to new rack-optimized connections, or added as direct connections (you can create new rack-optimized connections when adding EtherNet/IP I/O modules at runtime).
- You can add I/O modules until you reach the limits of the communication module:

1756-ENBT Maximums:	1756-ENET/B Maximums:
4500 pps	810 pps
64 TCP connections	64 TCP connections
128 CIP connected messages	160 CIP connected messages
128 connected bridged messages	128 connected bridged messages
32 CIP connected end node messages	32 CIP connected end node messages
256 CIP unconnected messages	64 CIP unconnected messages

For more information... The *EtherNet/IP Performance Application Guide*, ENET-AP001 provides guidelines on configuring an EtherNet/IP network to control I/O.

Determine When Data Is Updated

ControlLogix controllers update date asynchronous with the execution of logic. Use the following flowchart to determine when a producer (controller, input module, or bridge module) will send data.



Reconfigure an I/O Module

If an I/O module support reconfiguration, you can reconfigure the module via:

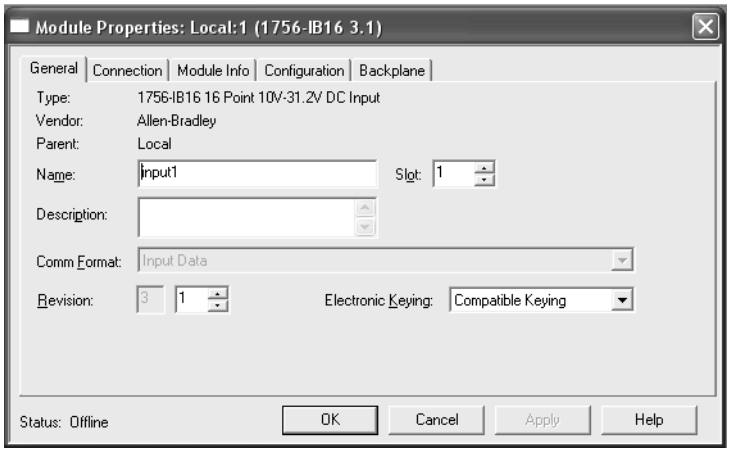
- Module Properties dialog in RSLogix 5000 software
- MSG instruction in program logic

WARNING

Use care when changing the configuration of an I/O module. You could inadvertently cause the I/O module to operate incorrectly.

Reconfigure a module via RSLogix 5000 software

To change the configuration of an I/O module via RSLogix 5000 software, highlight the module in the I/O Configuration tree. Right-click and select Properties.



Reconfigure a module via a MSG instruction

To change the configuration of an I/O module programmatically, use a MSG instruction of type Module Reconfigure to send new configuration information to an I/O module. During the reconfiguration:

- Input modules continue to send input data to the controller.
- Output modules continue to controller their output devices.

A Module Reconfigure message requires the following configuration properties:

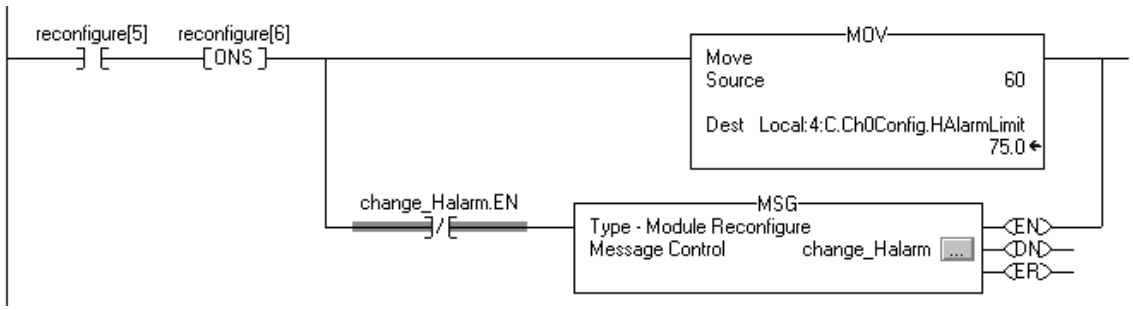
In this property:	Select:
Message Type	Module Reconfigure

To reconfigure an I/O module:

1. Set the required member of the configuration tag of the module to the new value.
2. Send a Module Reconfigure message to the module.

EXAMPLE Reconfigure an I/O module

When reconfigure[5] is on, the MOV instruction sets the high alarm to 60 for the local module in slot 4. The Module Reconfigure message then sends the new alarm value to the module. The ONS instruction prevents the rung from sending multiple messages to the module while the reconfigure[5] is on.



Notes:

Develop Applications

Use This Chapter

For this information:	See:
Manage Tasks	6-1
Organize Tags	6-4
Select a Programming Language	6-5
Monitor Controller Status	6-6
Monitor Connections	6-7

Manage Tasks

See:

- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

A Logix5000 controller lets you use multiple tasks to schedule and prioritize the execution of your programs based on specific criteria. This balances the processing time of the controller among the different operations in your application.

- The controller executes only one task at one time.
- A different task can interrupt a task that is executing and take control.
- In any given task, only one program executes at one time.

Defining tasks

A task provides scheduling and priority information for a set of one or more programs. You can configure tasks as continuous, periodic, or event. The ControlLogix controller supports as many as 32 tasks, only one of which can be continuous.

A task can have as many as 100 separate programs (including equipment phases), each with its own executable routines and program-scoped tags. Once a task is triggered (activated), all the programs assigned to the task execute in the order in which they are grouped. Programs can only appear once in the Controller Organizer and cannot be shared by multiple tasks.

Each task in the controller has a priority level. The operating system uses the priority level to determine which task to execute when multiple tasks are triggered. There are 15 configurable priority levels for periodic tasks that range from 1-15, with 1 being the highest priority and 15 being the lowest priority. A higher priority task will interrupt any lower priority task. The continuous task has the lowest priority and is always interrupted by a periodic or event task.

A ControlLogix controller supports three types of tasks:

Task:	Description:
continuous	The continuous task runs in the background. Any CPU time not allocated to other operations (such as motion, communications, and periodic or event tasks) is used to execute the programs within the continuous task.
periodic	A periodic task performs a function at a specific period. Whenever the time for the periodic task expires, the periodic task executes.
event	An event task performs a function only when a specific event (trigger) occurs. Whenever the trigger for the event task occurs, the event task executes.

Define programs

Each program contains program tags, a main executable routine, other routines, and an optional fault routine. Each task can schedule as many as 100 programs (including equipment phases).

The scheduled programs within a task execute to completion from first to last. Programs that aren't attached to any task show up as unscheduled programs. You must specify (schedule) a program within a task before the controller can scan the program.

Unscheduled programs within a task are downloaded to the controller with the entire project. The controller verifies unscheduled programs but the controller does not execute them.

Define routines

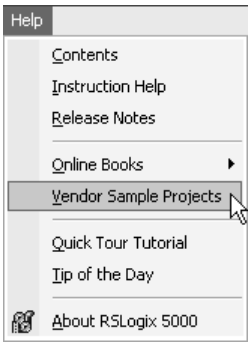
A routine is a set of logic instructions in a single programming language, such as ladder diagram. Routines provide the executable code for the project in a controller. A routine is similar to a program file or subroutine in a PLC or SLC processor.

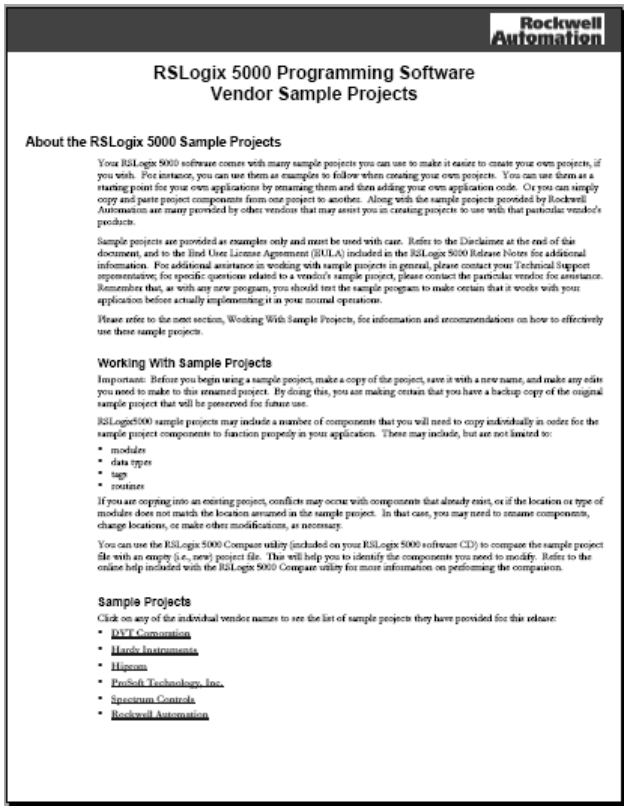
Each program has a main routine. This is the first routine to execute when the controller triggers the associated task and calls the associated program. Use logic, such as the JSR instruction, to call other routines.

You can also specify an optional program fault routine. The controller executes this routine if it encounters an instruction-execution fault within any of the routines in the associated program.

Sample controller projects

RSLogix 5000 Enterprise programming software includes sample projects that you can copy and then modify to fit your application. From RSlogix 5000 software, select Help → Vendor Sample Projects to display a list of available, sample projects.





Scroll down to here and select the appropriate set of sample projects

For more information... The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- select which task to use
- configure tasks
- prioritize tasks
- inhibit tasks

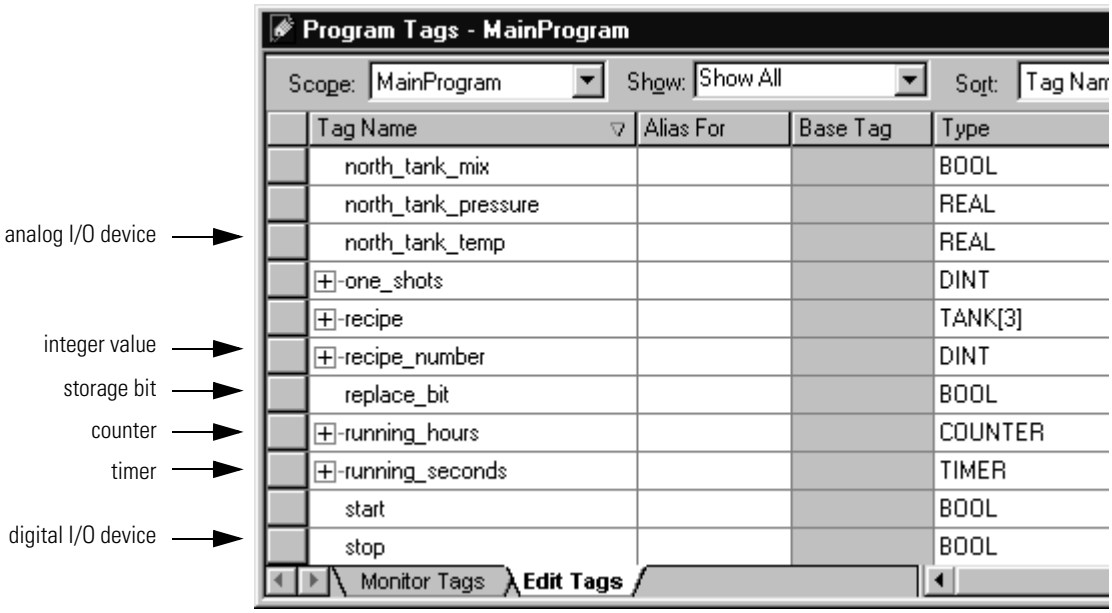
Organize Tags

See:

- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

With a Logix5000 controller, you use a tag (alphanumeric name) to address data (variables). In Logix5000 controllers, there is no fixed, numeric format. The tag name itself identifies the data. This lets you:

- organize your data to mirror your machinery
- document (through tag names) your application as you develop it



When you create a tag, you assign the following properties to the tag:

- tag type
- data type
- scope

For more information...

The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- define tags
- create tags, arrays, and user-defined structures
- address tags
- create aliases to tags
- assign indirect addresses

Select a Programming Language

The ControlLogix controller supports these programming languages, both online and offline:

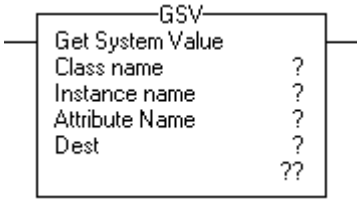
If you are programming:	Use this language:
continuous or parallel execution of multiple operations (not sequenced)	ladder diagram (LD)
boolean or bit-based operations	
complex logical operations	
message and communication processing	
machine interlocking	
operations that service or maintenance personnel may have to interpret in order to troubleshoot the machine or process	
continuous process and drive control	function block diagram (FBD)
loop control	
calculations in circuit flow	
high-level management of multiple operations	sequential function chart (SFC)
repetitive sequence of operations	
batch process	
motion control using structured text	
state machine operations	
complex mathematical operations	structured text (ST)
specialized array or table loop processing	
ASCII string handling or protocol processing	

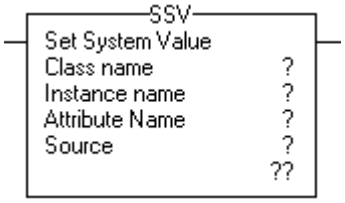
For more information... The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- design and program sequential function chart (SFC) logic
- program structured text (ST) logic
- program ladder diagram (LD) logic
- program function block diagram (FBD) logic
- force logic

The *Logix5000 Controllers Execution Time and Memory Use Reference Manual*, publication 1756-RM087 provides information on memory use and execution times for instructions.

Monitor Controller Status





The ControlLogix controller uses Get System Value (GSV) and Set System Value (SSV) instructions to get and set (change) controller data. The controller stores system data in objects. There is no status file, as in the PLC-5 processor.

The GSV instruction retrieves the specified information and places it in the destination. The SSV instruction sets the specified attribute with data from the source.

When you enter a GSV/SSV instruction, the programming software displays the valid object classes, object names, and attribute names for each instruction. For the GSV instruction, you can get values for all the available attributes. For the SSV instruction, the software displays only those attributes you are allowed to set.

In some cases, there will be more than one instance of the same type of object, so you might also have to specify the object name. For example, there can be several tasks in your application. Each task has its own TASK object that you access by the task name.

You can access these object classes:

- AXIS
- CONTROLLER
- CONTROLLERDEVICE
- CST
- DF1
- FAULTLOG
- MESSAGE
- MODULE
- MOTIONGROUP
- PROGRAM
- ROUTINE
- SERIALPORT
- TASK
- WALLCLOCKTIME

For more information...

The *Logix5000 Controllers General Instructions Reference Manual*, 1756-RM003 describes how to use the GSV and SSV instructions. These instructions support several different attributes of information.

The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- handle major faults
- handle minor faults
- determine controller memory use

Monitor Connections

See:

- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Design Considerations Reference Manual*, 1756-RM094

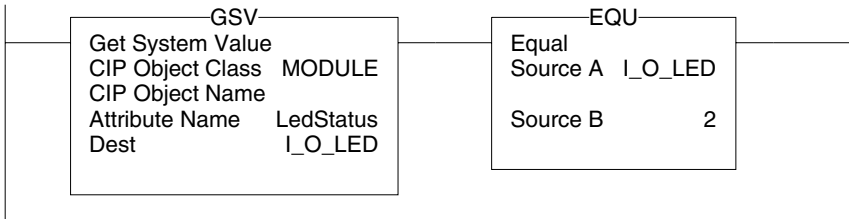
If communication with a device in the I/O configuration of the controller does not occur for 100 ms or 4 times the RPI (whichever is less), the communication times out and the controller produces the following warnings:

- The I/O LED on the front of the controller flashes green.
- A ⚠ shows over the I/O configuration folder and over the device (s) that has timed out.
- A module fault code is produced, which you can access through:
  - Module Properties dialog box for the module
  - GSV instruction

Determine if communication has timed out with any device

If communication times out with at least one device (module) in the I/O configuration of the controller, the I/O LED on the front of the controller flashes green.

- The GSV instruction gets the status of the I/O LED and stores it in the I_O_LED tag.
- If I_O_LED equal 2, the controller has lost communication with at least one device.



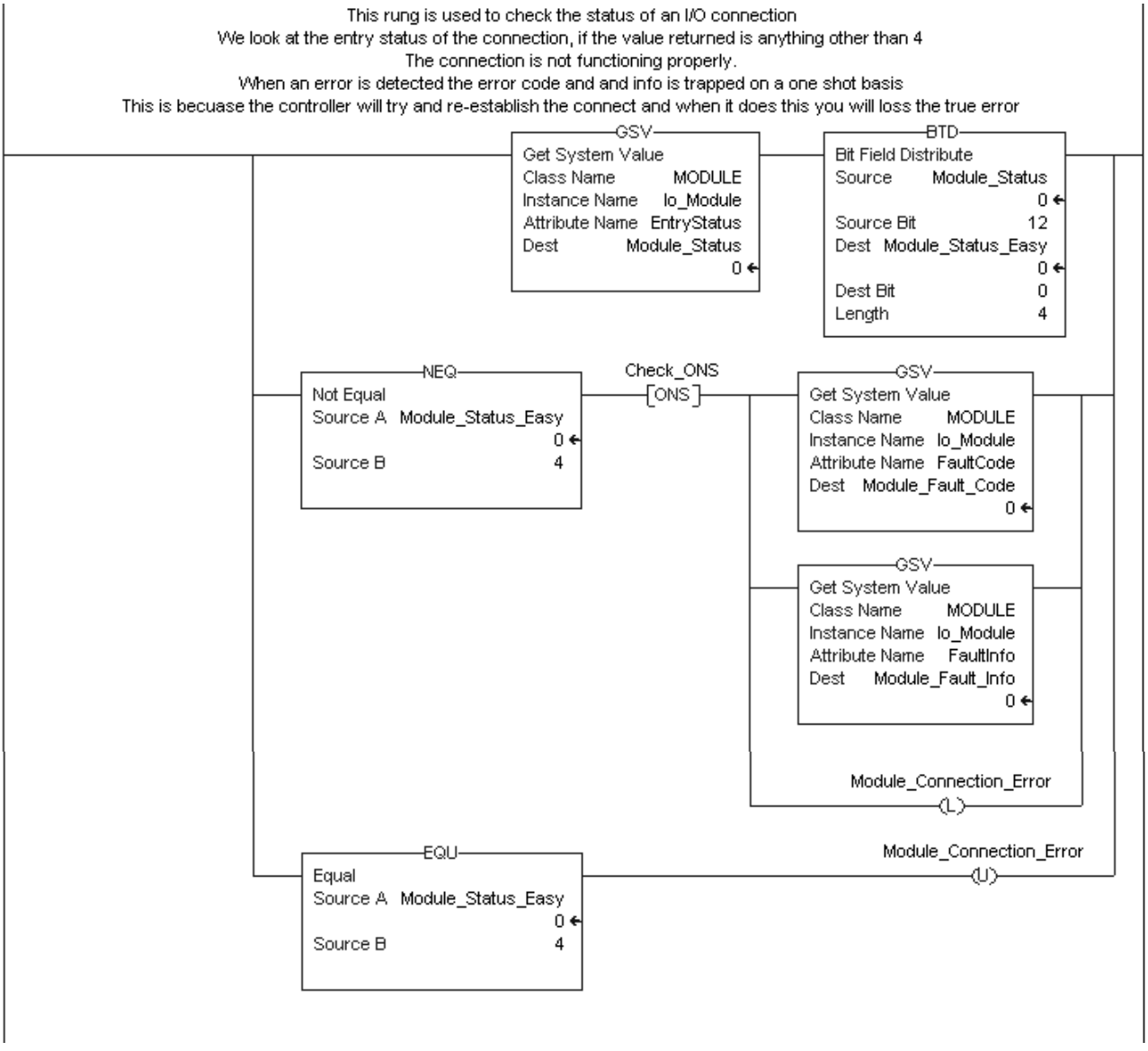
where:

I_O_LED is a DINT tag that stores the status of the I/O LED on the front of the controller.

Determine if communication has timed out with a specific I/O module

If communication times out with a device (module) in the I/O configuration of the controller, the controller produces a fault code for the module.

- The GSV instruction gets the fault code for Io_Module and stores it in the Module_Status tag.
- If Module_Status is any value other than 4, the controller is not communicating with the module.



Interrupt the execution of logic and execute the fault handler

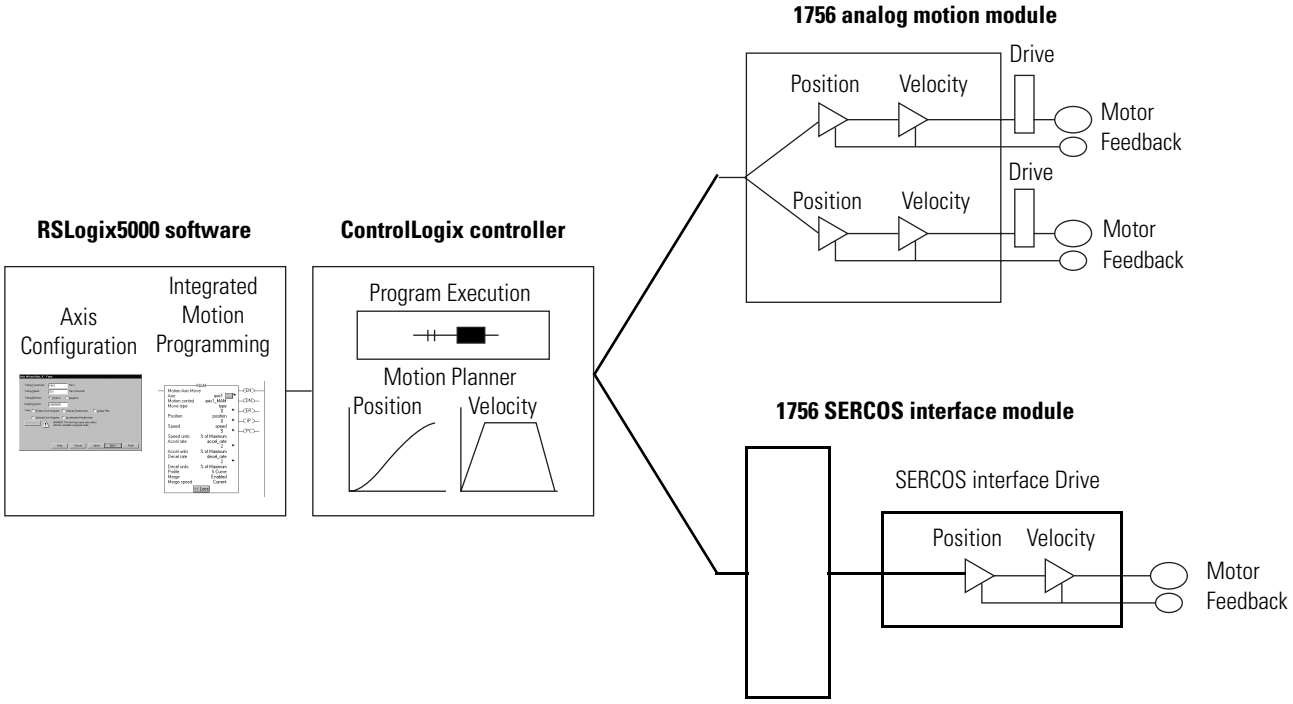
1. In the controller organizer, right-click the module and select *Properties*.
2. Click the Connection tab.
3. Select (check) the Major Fault If Connection Fails While in Run Mode check box.
4. Develop a routine for the Controller Fault Handler. See the *Logix5000 Controllers Common Procedures*, publication 1756-PM001.

Notes:

Configure Motion

Use This Chapter

The ControlLogix controller, 1756 motion modules, and RSLogix 5000 software create an integrated motion control system.



Use this chapter to set up and program motion control. If you aren't using SERCOS interface drives and modules, skip actions 3 and 4.

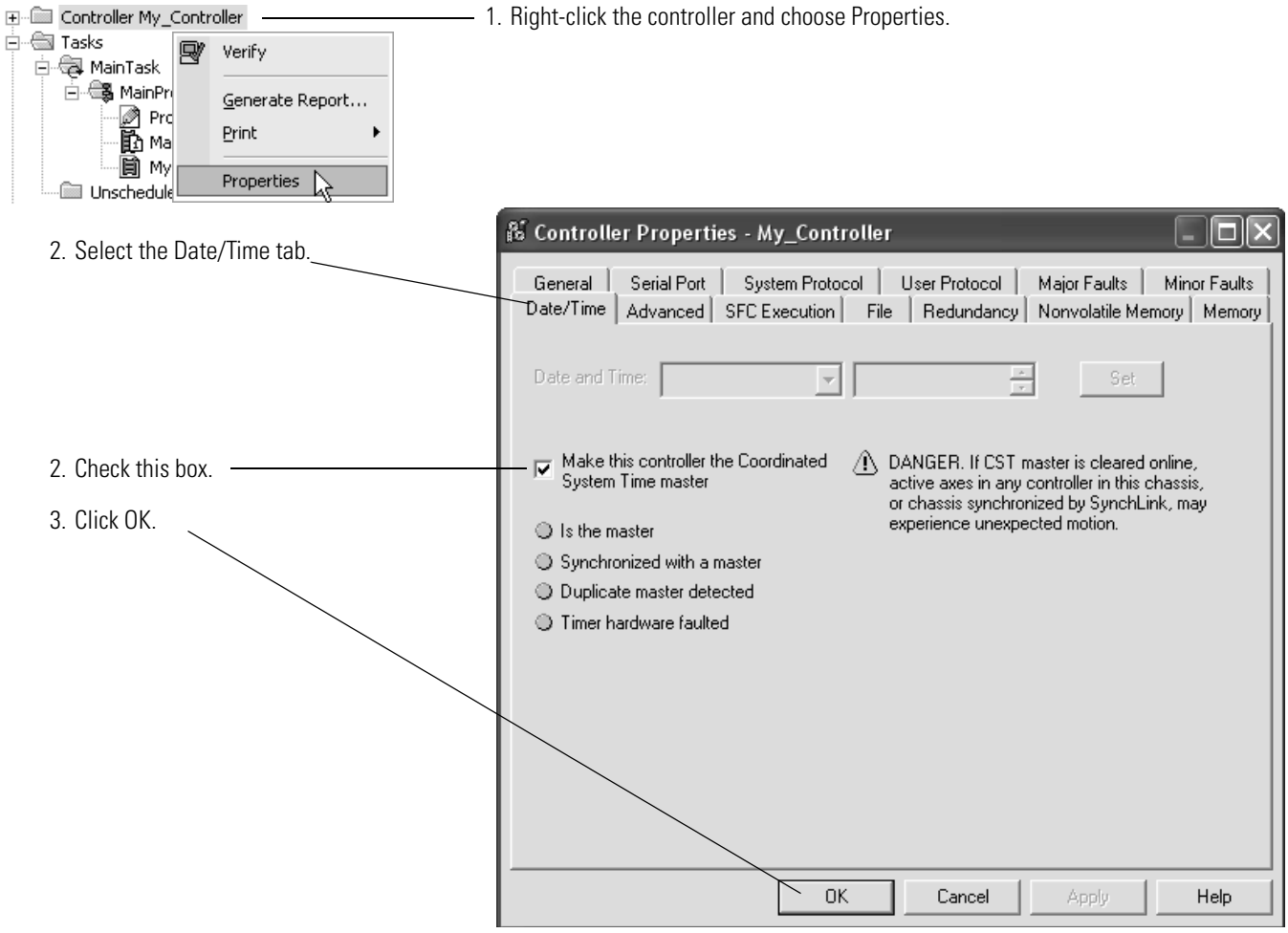
Action	See page
1. Make the Controller the CST Master	7-2
2. Add the Motion Modules	7-3
3. Add SERCOS interface Drives	7-4
4. Set Up Each SERCOS Interface Module	7-5
5. Add the Motion Group	7-7
6. Add Your Axes	7-9
7. Set Up Each Axis	7-10
8. Check the Wiring of Each Drive	7-13
9. Tune Each Axis	7-14
10. Program Motion Control	7-15
11. Additional Actions	7-17

Make the Controller the CST Master

You must make one module in the chassis the master clock for motion control. This module is called the coordinated system time (CST) master.

coordinated system time (CST) master	The master clock for motion control for a chassis. The motion modules set their clocks to the master.

In most cases, make the controller the CST master.



If you have more than 1 controller in the chassis

If you have more than 1 controller in the chassis, choose 1 of the controllers to be the CST master. You can't have more than one CST master for the chassis.

Add the Motion Modules

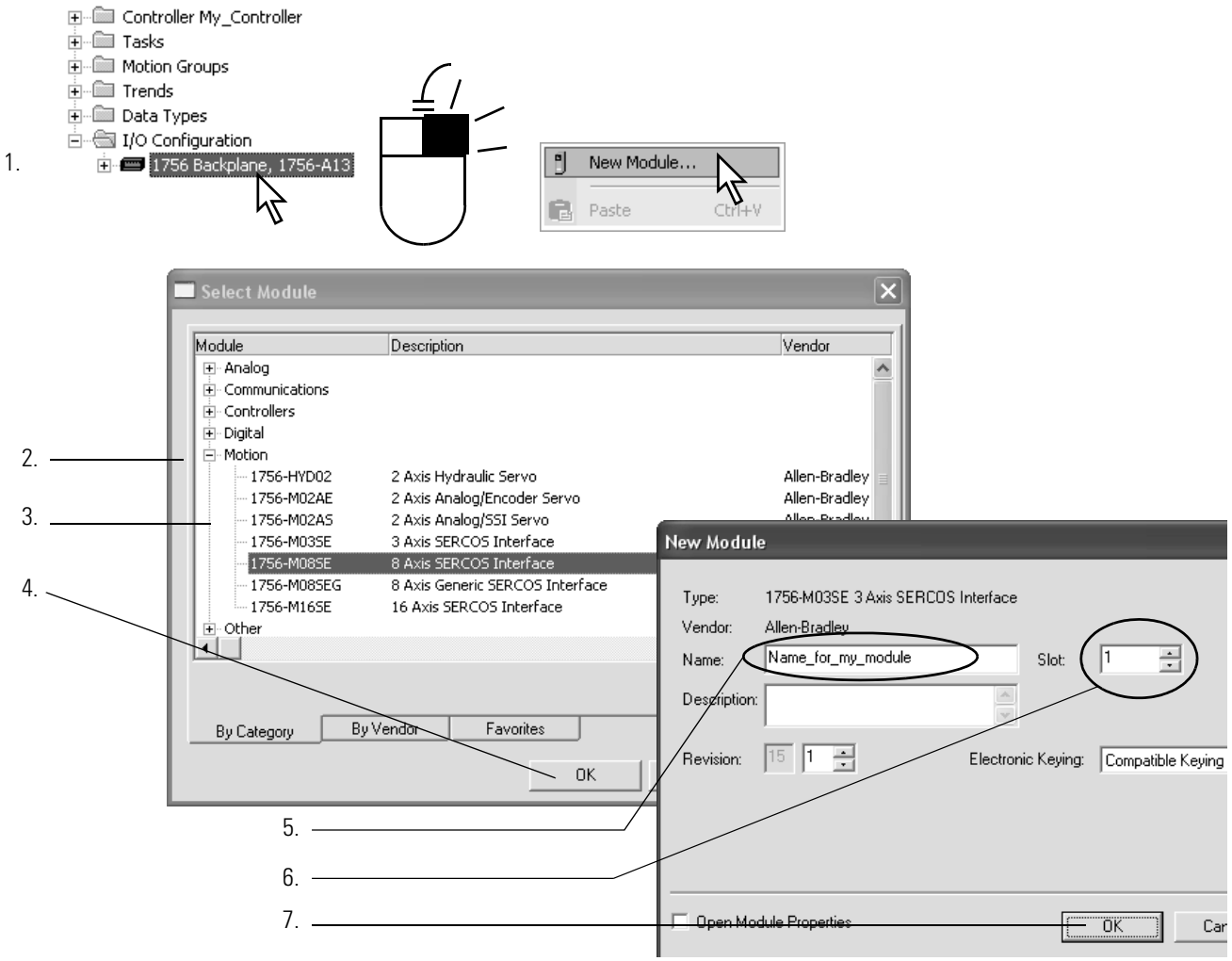
See:

- *Motion Analyzer*, PST-SG003
- *ControlLogix Selection Guide*, 1756-SG001

Each ControlLogix controller controls up to 16 motion modules:

IMPORTANT For your motion modules, use the firmware revision that goes with the firmware revision of your controller. See the release notes for your controller firmware to see which firmware you need for your motion modules.

If your equipment uses:	And this feedback:	Use this motion module:
Rockwell Automation SERCOS interface drives	⇒ ⇒ ⇒ ⇒	1756-M03SE (3 axes) 1756-M08SE (8 axes) 1756-M16SE (16 axes) 1756-L60M03SE (3 axes)
analog command signal	quadrature feedback	1756-M02AE
	LDT feedback	1756-HYD02
	SSI feedback	1756-M02AS



Add SERCOS interface Drives

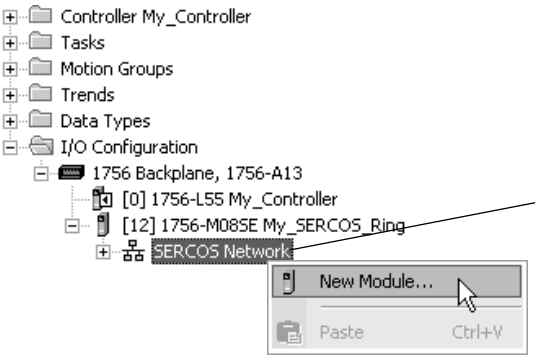
See:

- *Motion Analyzer*, PST-SG003
- *ControlLogix Selection Guide*, 1756-SG001
- *Logix5000 Motion Module User Manual*, 1756-UM006

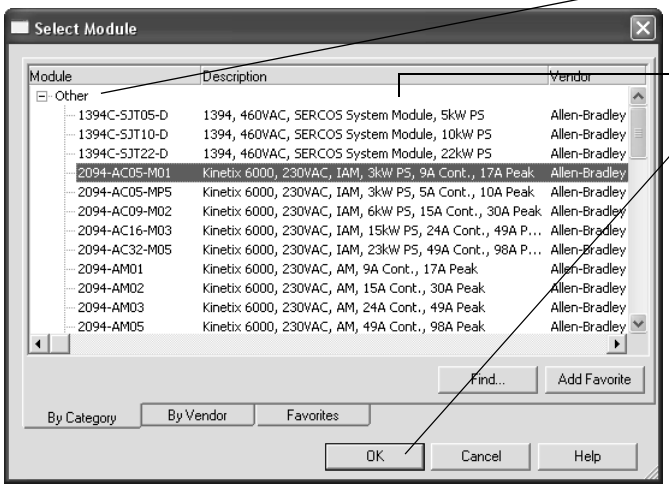
Choose from these SERCOS interface drives:

- 1394
- Kinetix 6000
- Ultra3000
- 8720MC

Add SERCOS interface drives to the I/O configuration of the controller. This lets you use RSLogix 5000 software to set up the drives.



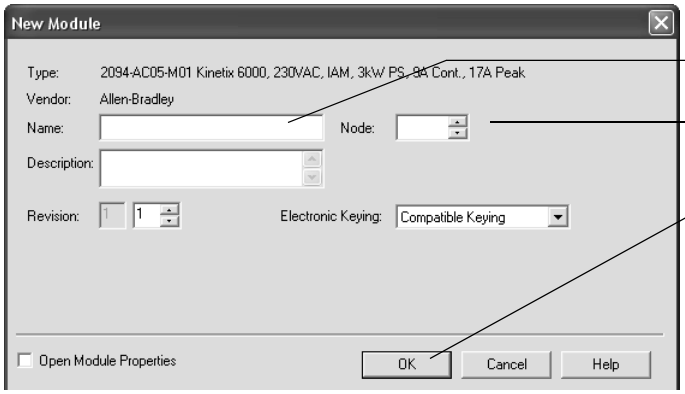
1. Right-click the SERCOS network and select *New Module*.



2. Open the Other category.

3. Select your drive.

4. Click OK.



5. Type a name for the drive.

6. Select the node number of the drive on the SERCOS ring.

7. Click OK.

Set Up Each SERCOS Interface Module

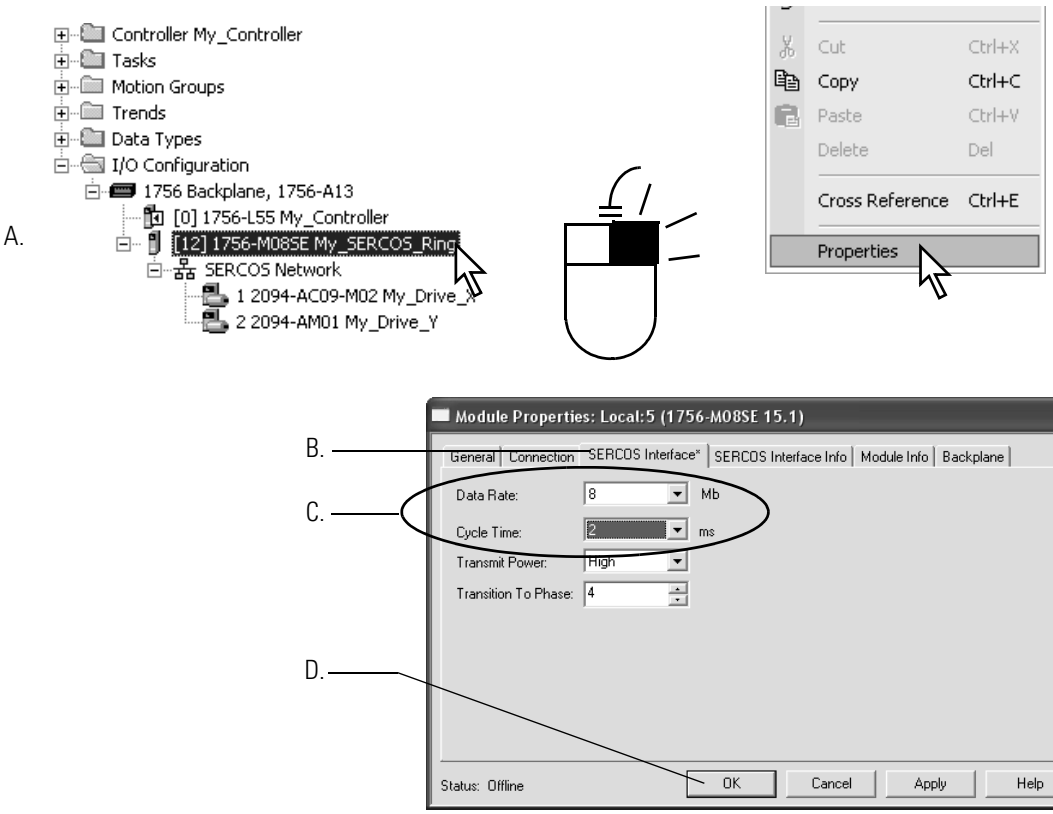
Set the data rate and cycle time for each SERCOS interface module in your project.

Action	Details
1. **Decide which data rate to use.**	Do your drives have a 8 Mb data rate (most do)? • YES — Use a 8 MB data rate. • NO — Use a 4 MB data rate.
2. **Decide which cycle time to use.**	Use the following table to decide the cycle time for your SERCOS interface module:

Data rate	Type of drives	Number of drives on the ring	Cycle time
4 Mb	Kinetix 6000	1 or 2	0.5 ms
		3 or 4	1 ms
		5…8	2 ms
		9…16	Can't do. You must have 2 motion modules.
	NOT Kinetix 6000	1…4	1 ms
		5…8	2 ms
		9…16	Can't do. You must have 2 motion modules.
8 Mb	Kinetix 6000	1…4	0.5 ms
		5…8	1 ms
		9…16	2 ms
	NOT Kinetix 6000	1…8	1 ms
		9…16	2 ms

Action	Details

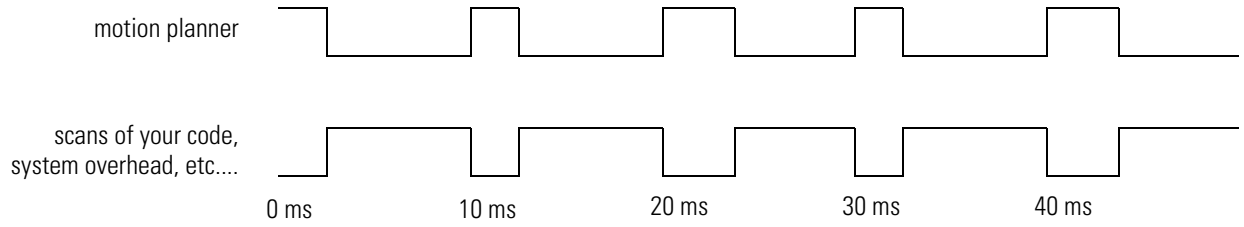
3. **Set the data rate and cycle time.**



Add the Motion Group

Add a motion group to set up the motion planner.

motion planner	Part of the controller that takes care of position and velocity information for your axes
coarse update period	How often the motion planner runs. When the motion planner runs, it interrupts all other tasks regardless of their priority.



In this example, the coarse update period = 10 ms. Every 10 ms the controller stops scanning your code and whatever else it is doing and runs the motion planner.

IMPORTANT Add only 1 motion group for the project. RSLogix 5000 software doesn't let you add more than 1 motion group.

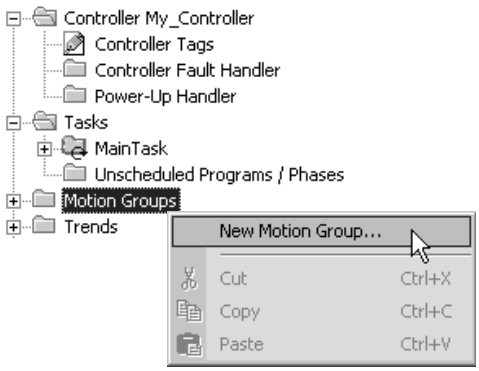
Action	Details
1. **Decide which coarse update period to use.**	The coarse update period is a trade-off between updating positions of your axes and scanning your code. Use this table as a rough starting point.

If you have	And you have	Then use this coarse update period
less than 11 axes	⇒ ⇒ ⇒ ⇒	10 ms
11 axes or more	**NO** SERCOS interface modules that use a 2 ms cycle time	1 ms per axis
	SERCOS interface modules that use a 2 ms cycle time	1 ms per axis rounded up to an even number

Action	Details

2. **Add the motion group.**

A. Right-click the Motion Groups folder and select New Motion Group

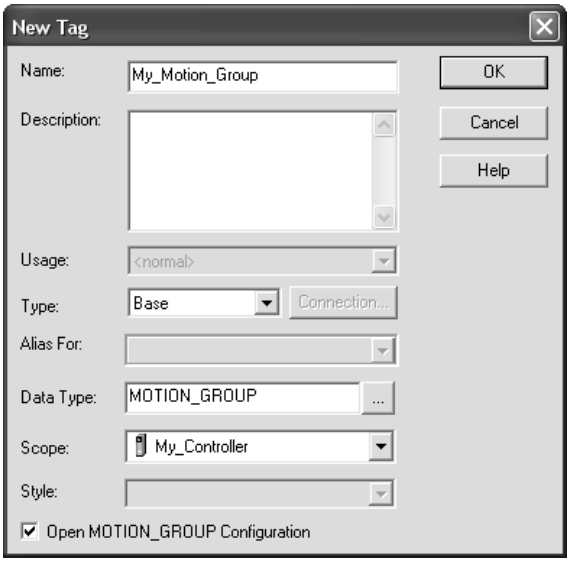


B. Type a name for the motion group.

C. Check this box.

D. Click OK.

The Motion Group Wizard opens.

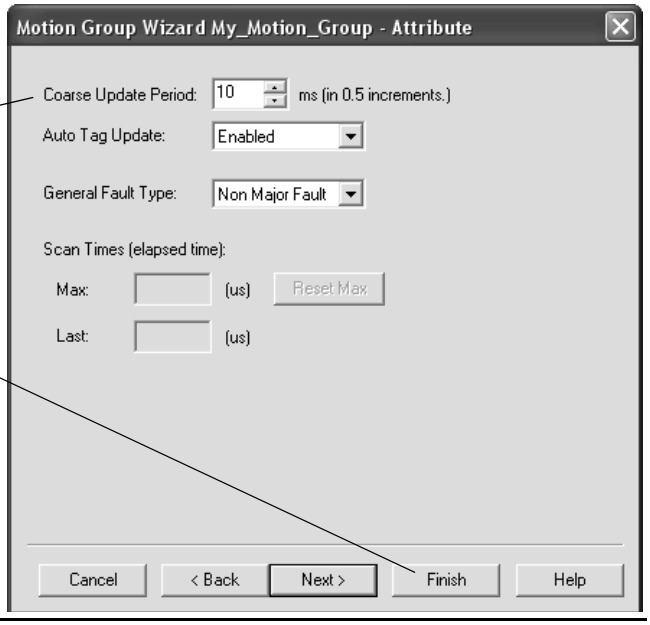


3. **Set the coarse update period.**

A. Click Next one time.

B. Type the coarse update period from Step 1.

C. Click *Finish*.



Add Your Axes

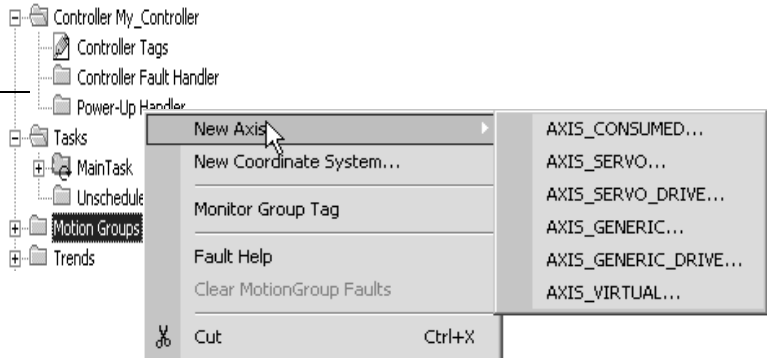
Add an axis for each of your drives.

Action	Details
1. **Decide which data type to use.**	Use the following table to decide which data type to use for an axis.

If you this motion module for the axis	Then plan to use this data type
1756-M03SE 1756-M08SE 1756-M16SE 1756-L60M03SE	AXIS_SERVO_DRIVE
1756-M08SEG	AXIS_GENERIC_DRIVE
1756-M02AE 1756-HYD02 1756-M02AS	AXIS_SERVO

2. **Add an axis.**

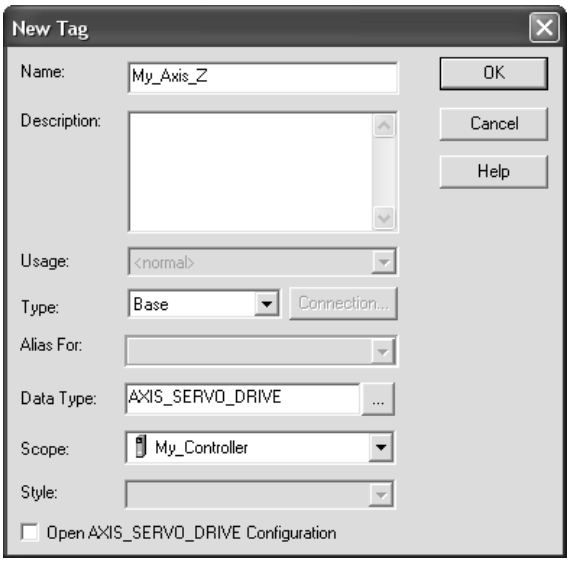
A. Right-click your motion group, select New Axis, and choose the data type for the axis.



B. Type a name for the axis.

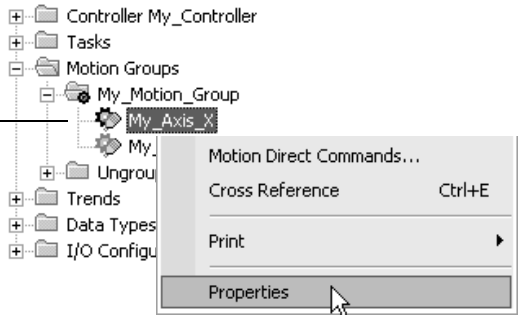
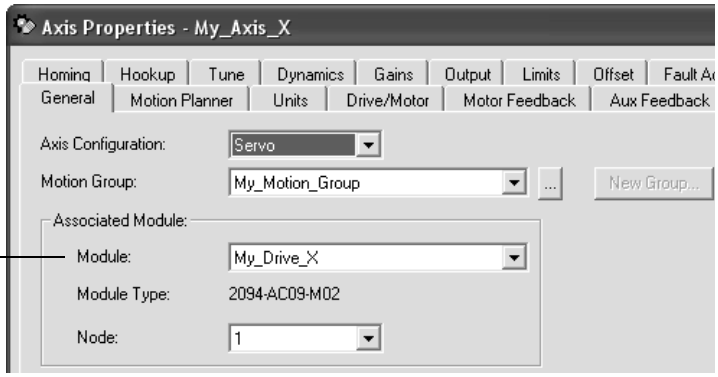
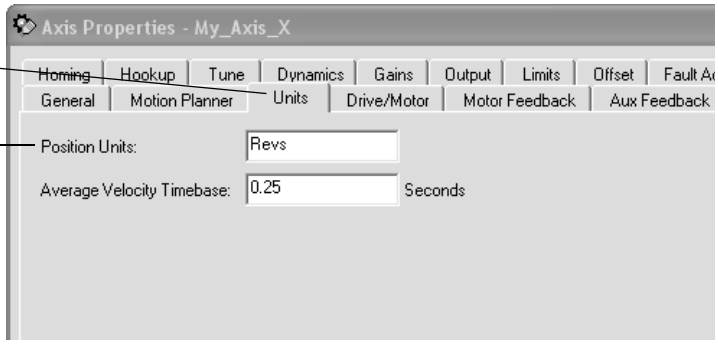
C. Uncheck this box.

D. Click OK.



Set Up Each Axis

The following steps show how to set up the axis of a SERCOS interface drive. The steps are slightly different if you have a different type of drive.

Action	Details
1. **Open the properties for the axis.**	Right-click the axis and select Properties. 
2. **Select the drive for the axis.**	Select the name that you gave to the drive for this axis. 
3. **Set the units that you want to program in.**	A. Select the Units tab. B. Type the units that you want to use for programming, such as revs, degrees, inches, or millimeters. 

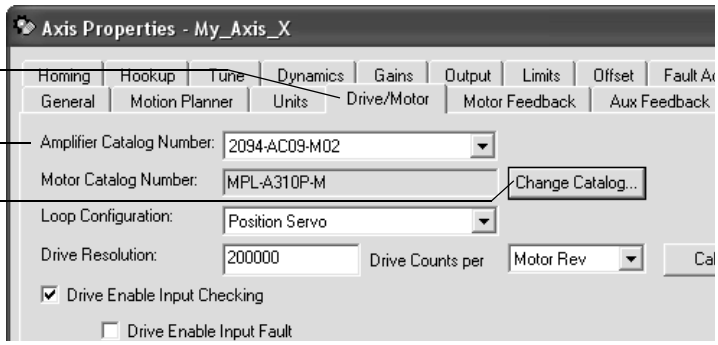
Action	Details

4. **Select the drive and motor catalog numbers.**

A. Select the Drive/Motor tab.

B. Select the catalog number of the drive.

C. Select the catalog number of the motor.



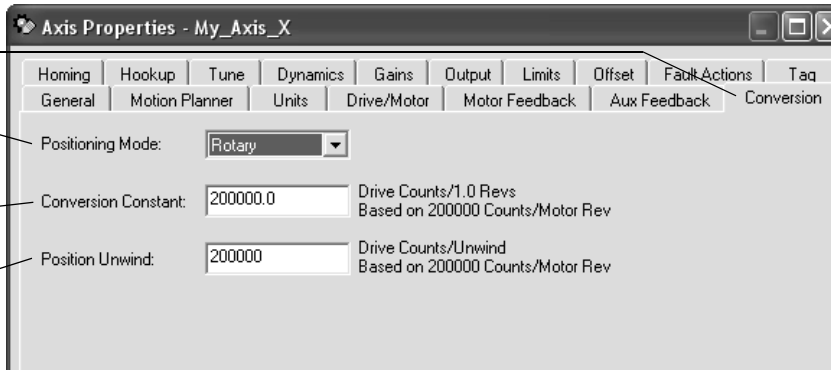
5. **Set the conversion between drive counts and units.**

A. Select the Conversion tab.

B. Select whether this is a rotary or linear axis.

C. Type the number of drive counts that equal one unit from Step 3B.

D. If this is a rotary axis, type the number of drive counts that you want to unwind after.

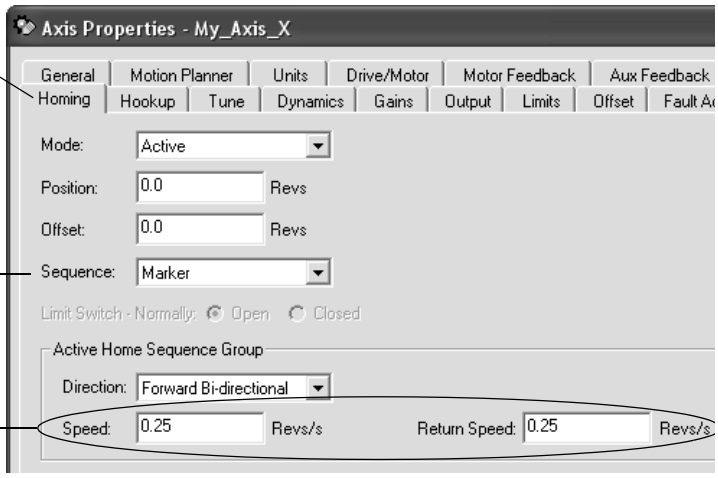


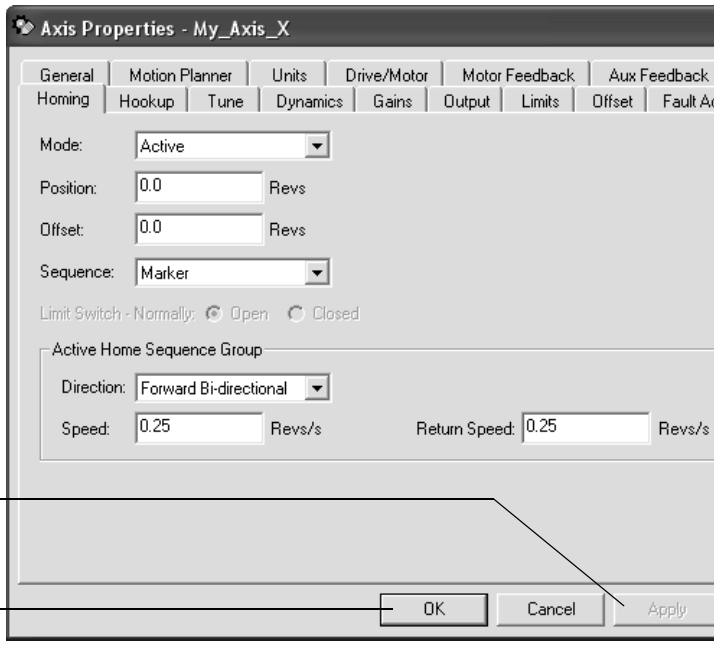
6. **Set up the homing sequence.**

A. Select the Homing tab.

B. Select the type of homing sequence that you want.

C. Type homing speeds.



Action	Details
7. **Apply your changes.** A. Click Apply. B. Click OK.	

Check the Wiring of Each Drive

See:

- *Logix5000 Motion Module User Manual*, 1756-UM006

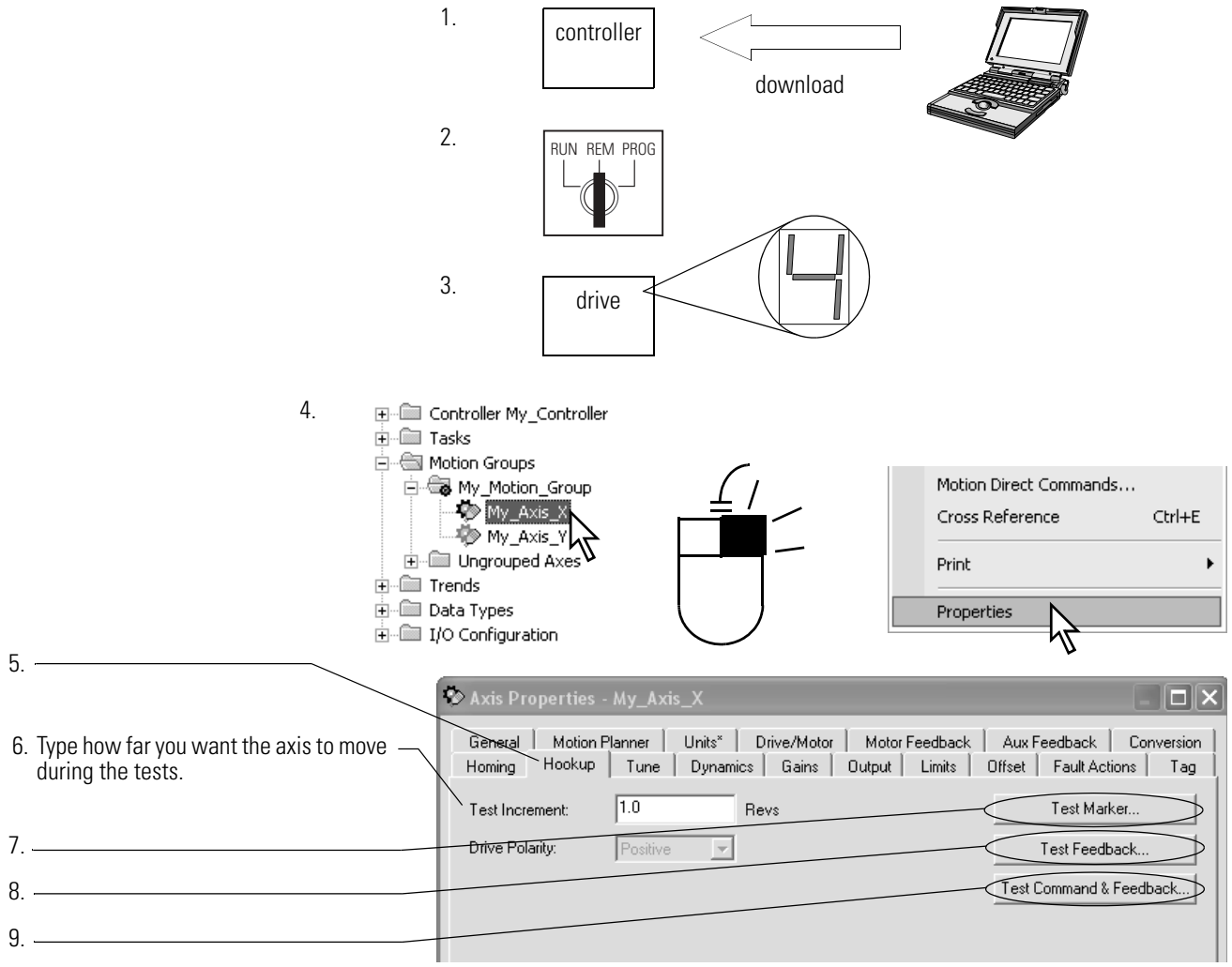
Use the hookup tests to check the wiring of a drive.

This test	Does this
Test marker	Checks that the drive gets the marker pulse.
Test feedback	Checks the polarity of the feedback.
Test command and feedback	Checks the polarity of the drive.

ATTENTION

These tests make the axis move even with the controller in remote program mode.

- Before you do the tests, make sure no one is in the way of the axis.
- Do not change the polarity after you do the tests. Otherwise you may cause an axis-runaway condition.



Tune Each Axis

See:

- *Logix5000 Motion Module User Manual*, 1756-UM006

Use the Tune tab to tune an axis.

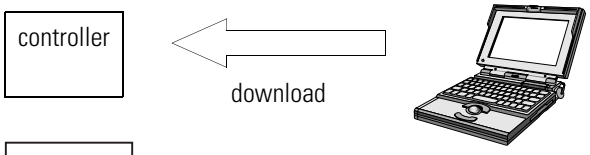
ATTENTION

When you tune an axis, it moves even with the controller in remote program mode. In that mode, your code is ***not*** in control of the axis.

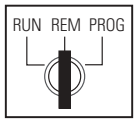
Before you tune an axis, make sure no one is in the way of the axis.

The default tuning procedure tunes the proportional gains. Typically, tune the proportional gains first and see how your equipment runs.

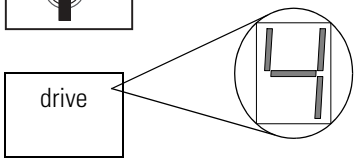
1.



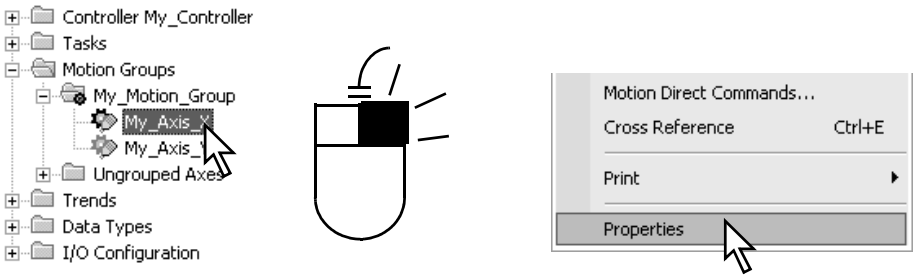
2.



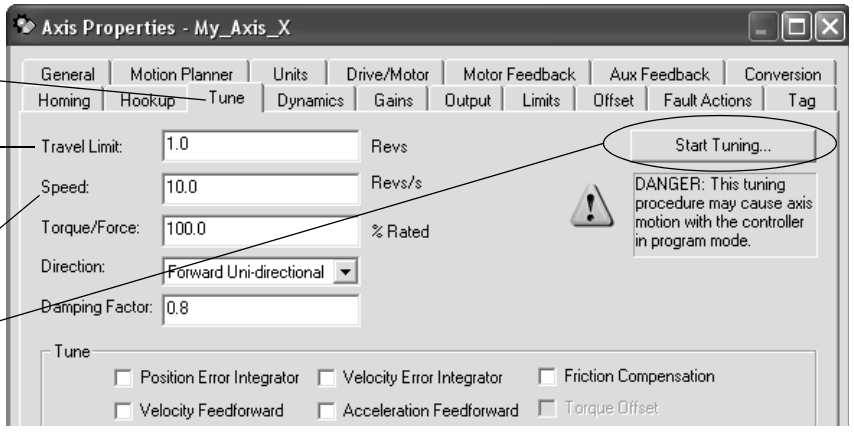
3.



4.



5. 6. 7. 8.



5. Tune

6. Type the limit of movement for the axis during the tuning procedure.

7. Type the maximum speed for the tuning procedure.

8. Start Tuning...

To tune the integral gains or feedforward, see *Logix5000 Motion Modules User Manual*, publication 1756-UM006.

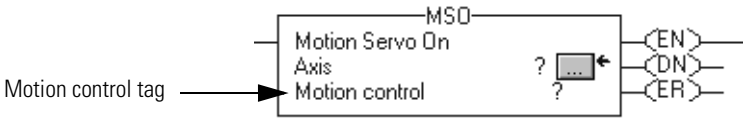
Program Motion Control

See:

- *Logix5000 Motion Module User Manual*, 1756-UM006
- *Logix5000 Controllers Common Procedures Manual*, 1756-PM001
- *Logix5000 Controllers Motion Instructions Reference Manual*, 1756-RM007
- *Logix5000 Controllers General Instructions Reference Manual*, 1756-RM003

The controller gives you a set of motion control instructions for your axes.

- Uses these instructions just like the rest of the Logix5000 instructions. You can program motion control in these programming languages:
  - ladder diagram (LD)
  - structured text (ST)
  - sequential function chart (SFC)
- Each motion instruction works on one or more axes.
- Each motion instruction needs a motion control tag. The tag uses a MOTION_INSTRUCTION data type. The tag stores the status information of the instruction.



ATTENTION

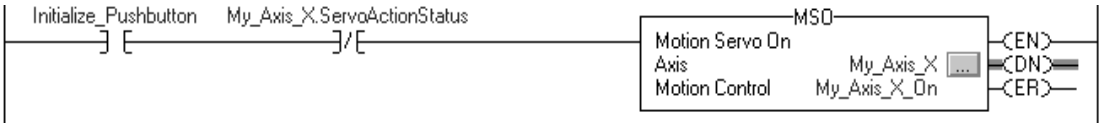
Use the tag for the motion control operand of motion instruction only once. Unintended operation of the control variables may happen if you re-use of the same motion control tag in other instructions.

Example

Here's an example of a simple ladder diagram that homes, jogs, and moves an axis.

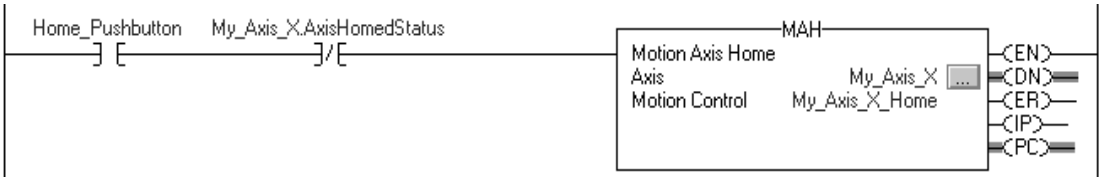
If Initialize_Pushbutton = on and the axis = off (My_Axis_X.ServoActionStatus = off) then

The MSO instruction turns on the axis.



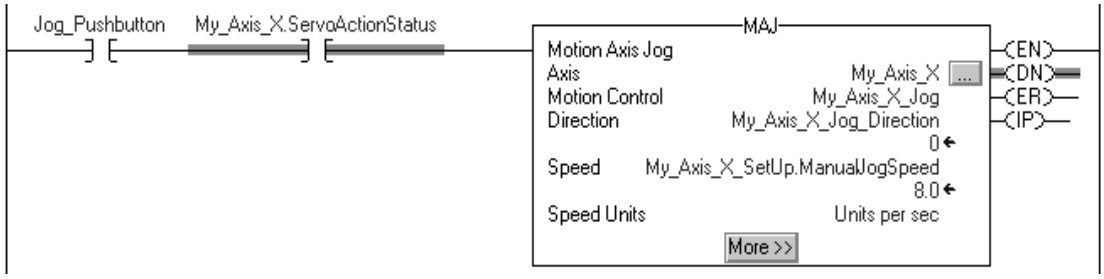
If Home_Pushbutton = on and the axis hasn't been homed (My_Axis_X.AxisHomedStatus = off) then

The MAH instruction homes the axis.



If Jog_Pushbutton = on and the axis = on (My_Axis_X.ServoActionStatus = on) then

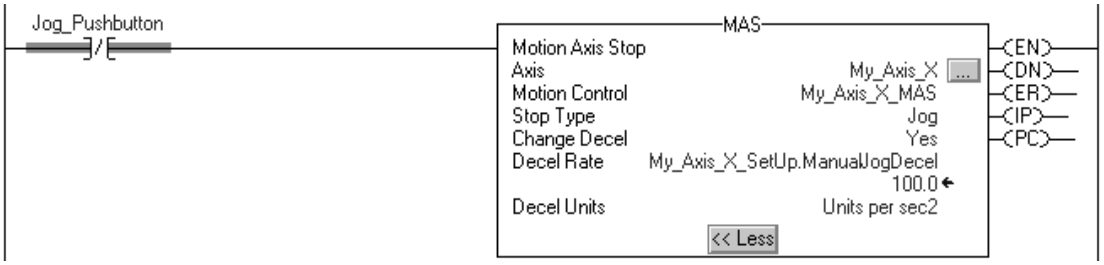
The MAJ instruction jogs the axis forward at 8 units/s.



If Jog_Pushbutton = off then

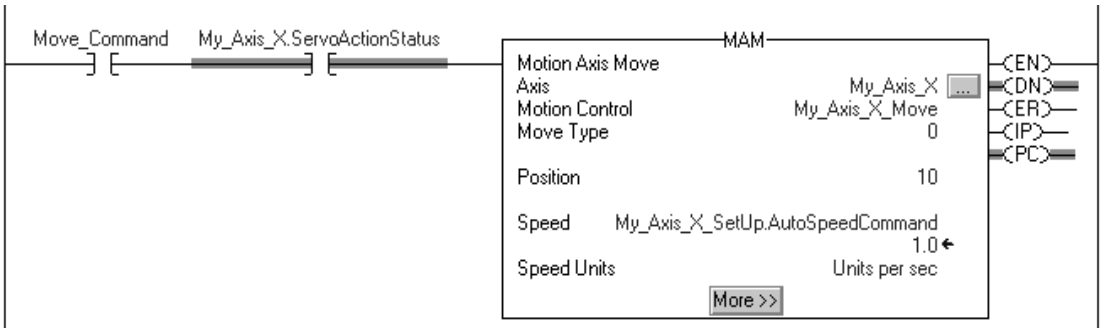
The MAS instruction stops the axis at 100 units/s^2

Make sure that Change Decel is Yes. Otherwise, the axis decelerates at it maximum speed.



If Move_Command = on and the axis = on (My_Axis_X.ServoActionStatus = on) then

The MAM instruction moves the axis. The axis move to the position of 10 units at 1 unit/s.



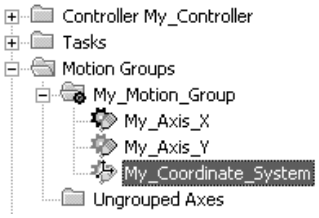
Additional Actions

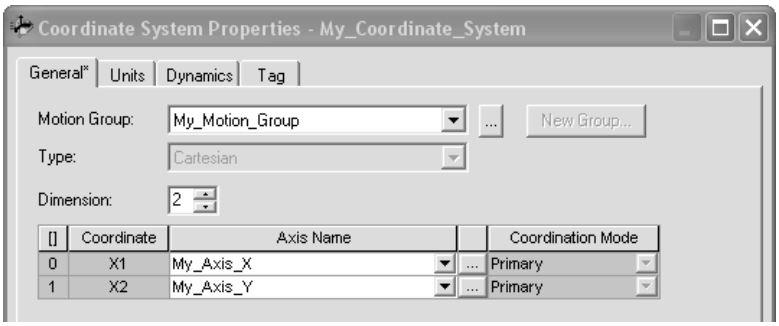
The following actions are optional and depend on your situation.

See:

- *Logix5000 Motion Module User Manual*, 1756-UM006
- *Logix5000 Controllers Motion Instructions Reference Manual*, 1756-RM007
- *Logix5000 Controllers General Instructions Reference Manual*, 1756-RM003

Action	Details
Set up a coordinate system	A coordinate system lets you interpolate circular or linear moves using coordinate points. Set up the coordinate in either 1, 2, or 3 dimensions.





Get status information

Use these methods to read motion status and configuration parameters in your code.

Method:	Example:
Read the MOTION_GROUP and AXIS tags	• Axis faults • Actual position of an axis • Motion status
Use a Get System Value (GSV) instruction	Actual position

Change configuration parameters

Use a Set System Value (SSV) instruction to write code that changes motion parameters. For example, you can change position loop gain, velocity loop gain, and current limits within your code.

Action	Details
Handle motion faults	The controller has these types of motion faults:

Type	Description	Example
Instruction error	Caused by a motion instruction: • Instruction errors do not impact controller operation. • Look at the error code in the motion control tag to see why an instruction has an error. • Fix instruction errors to optimize execution time and make sure that your code is accurate	A Motion Axis Move (MAM) instruction with a parameter out of range
Fault	Caused by a problem with the servo loop: • You choose whether or not motion faults give the controller major faults. • Can shutdown the controller if you do not correct the fault condition	• Loss of feedback • Actual position exceeding an overtravel limit

Configure PhaseManager

Use This Chapter

See:
- PhaseManager User Manual, LOGIX-UM001

The PhaseManager option of RSLogix 5000 software gives you a state model for your equipment. This chapter summarizes:

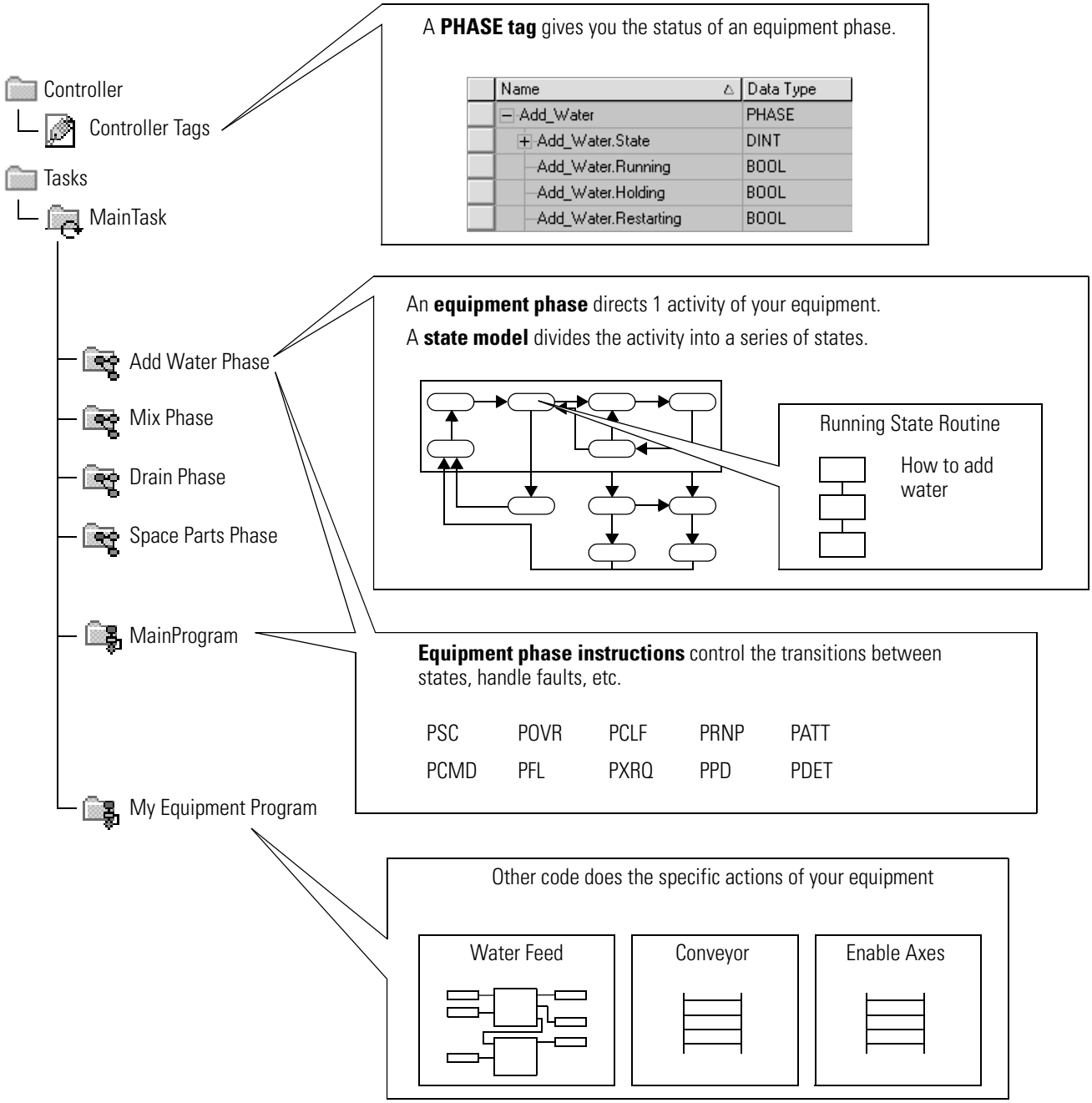
For this information:	See:
PhaseManager Overview	8-1
State Model Overview	8-3
Compare PhaseManager to Other State Models	8-6
Minimum System Requirements	8-6
Equipment Phase Instructions	8-7

PhaseManager Overview

PhaseManager lets you add equipment phases to your controller. An equipment phase helps you lay-out your code in sections that are easier to write, find, follow, and change.

Term	Description
equipment phase	An equipment phase is similar to a program: • You run the equipment phase in a task. • You give the equipment phase a set of routines and tags. An equipment phase is different from a program in these ways: • The equipment phase runs by a state model. • You use an equipment phase to do 1 activity of your equipment.
state model	A state model divides the operating cycle of your equipment into a series of states. Each state is an instant in the operation of the equipment. It's the actions or conditions of the equipment at a given time. The state model of an equipment phase is similar to the S88 and Pack*ML* state models.
state machine	An equipment phase includes an embedded state machine that: • calls the appropriate routine (state routine) for an acting state • manages the transitions between states with minimal coding You code the transition conditions. When the conditions are true, the equipment phase transitions the equipment to the next required state. • makes sure that the equipment goes from state to state along an allowable path
PHASE tag	When you add an equipment phase, RSLogix 5000 software makes a tag for the equipment phase. The tag uses the PHASE data type.

Here's how the PhaseManager into RSLogix 5000 programming software:



State Model Overview

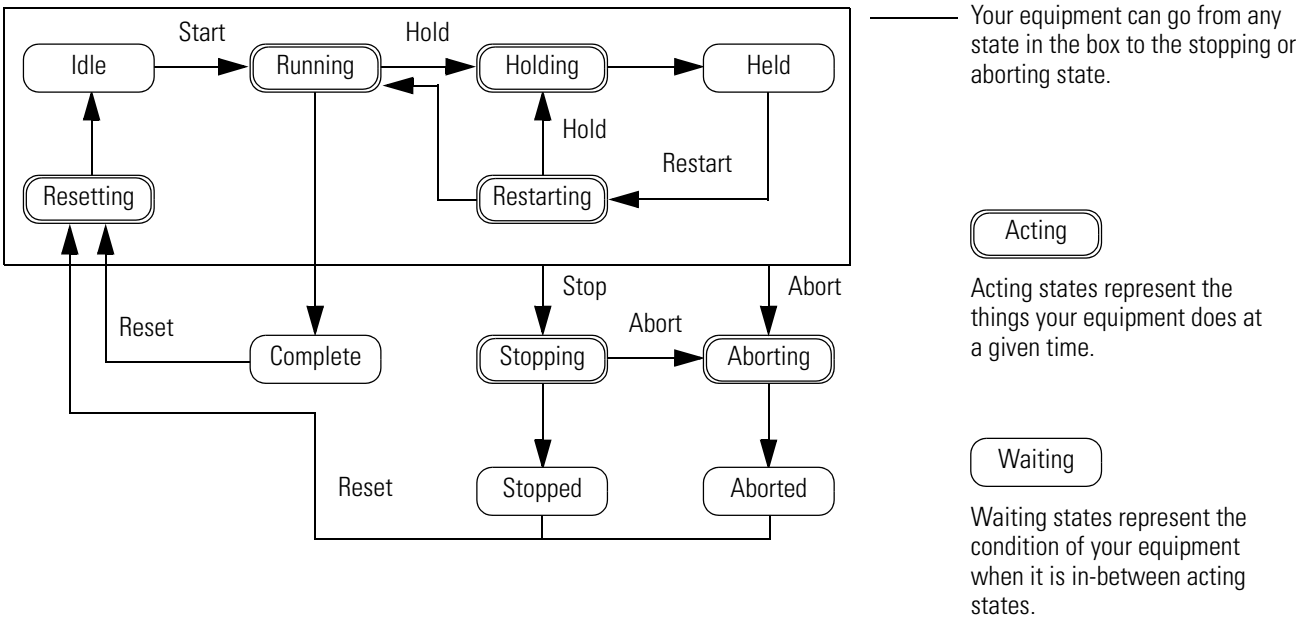
A state model divides the operating cycle of your equipment into a series of states. Each state is an instant in the operation of the equipment. It's the actions or conditions of the equipment at a given time.

In a state model, you define what your equipment does under different conditions, such as run, hold, stop, etc. You don't need to use all the states for your equipment. Use only the states that you want.

There are 2 types of states:

Type of state	Description
Acting (transition)	Does something or several things for a certain time or until certain conditions are met. An acting state runs one time or repeatedly.
Waiting (stable)	Shows that certain conditions are met and the equipment is waiting for the signal to go to the next state.

PhaseManager uses the following states:



With a state model, you define the behavior of your equipment and put it into a brief functional specification. In this way you show what happens and when it happens.

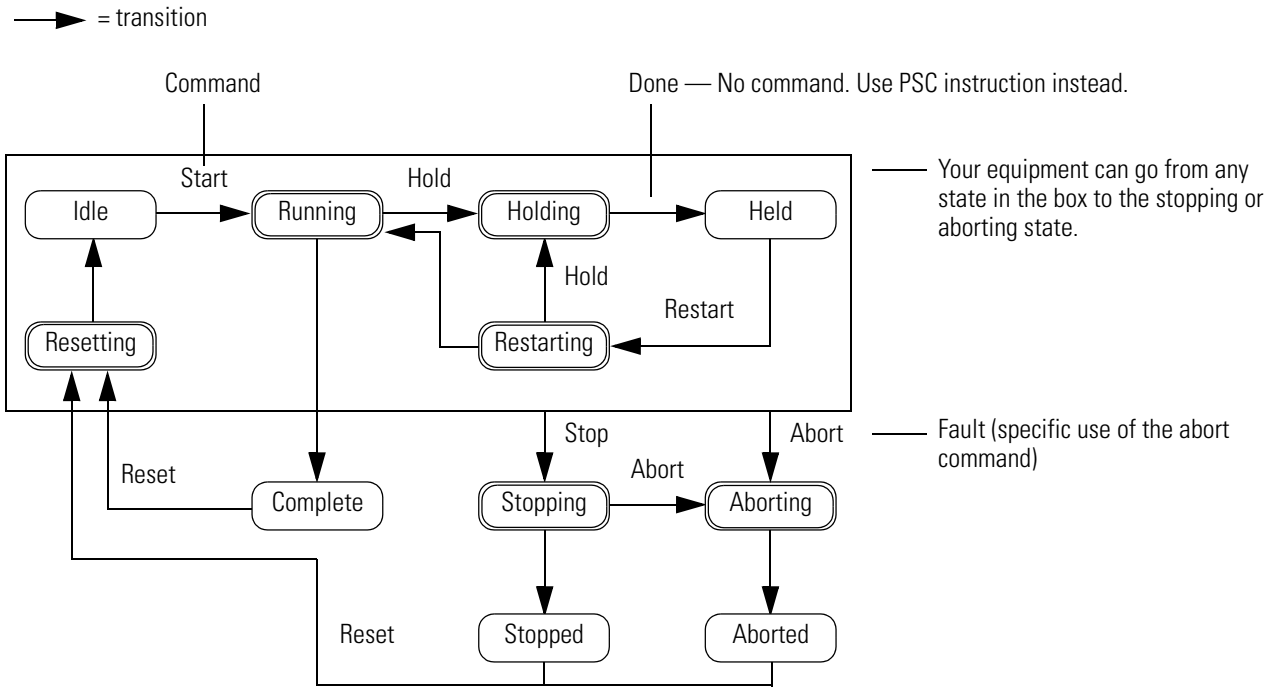
For this State:	Ask:
Stopped	What happens when you turn on power?
Resetting	How does the equipment get ready to run?
Idle	How do you tell that the equipment is ready to run?
Running	What does the equipment do to make product?
Holding	How does the equipment ***temporarily*** stop making product without making scrap?
Held	How do you tell if the equipment is safely holding?
Restarting	How does the equipment resume production after holding?
Complete	How do you tell when the equipment is done with what it had to do?
Stopping	What happens during an normal shutdown?
Aborting	How does the equipment shutdown if a fault or failure happens?
Aborted	How do you tell if the equipment is safely shutdown?

How equipment changes states

The arrows in the state model show to which states your equipment can go from the state it is in now.

- Each arrow is called a transition.
- A state model lets the equipment make only certain transitions. This gives the equipment the same behavior as any other equipment that uses the same model.

PhaseManager uses the following transitions:

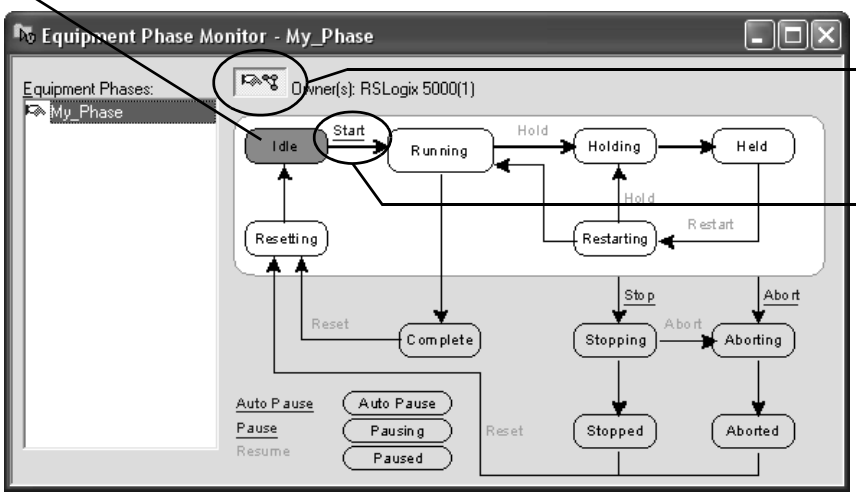


Type of transition	Description
Command	A command tells the equipment to start doing something or do something different. For example the operator pushes the start button to start production and the stop button to shutdown. PhaseManager uses these commands: reset stop restart start hold abort
Done	Equipment goes to a waiting state when it's done with what it's doing. You don't give the equipment a command. Instead, you set up your code to signal when the equipment is done. The waiting state shows that the equipment is done.
Fault	A fault tells you that something out of the ordinary has happened. You set up your code to look for faults and take action if it finds any. Suppose you want your equipment to shut down as fast as possible if a certain fault happens. In that case, set up your code look for that fault and give the abort command if it finds it.

Manually change states

RSLogix 5000 software has a window that lets you monitor and command an equipment phase.

State that the equipment phase is in right now



To manually change states:

1. Take ownership of the equipment phase.
2. Give a command.

Compare PhaseManager to Other State Models

This table compares PhaseManager's state model to other common state models:

S88	Pack*ML*	PhaseManager
Idle	Starting ⇒ Ready	Resetting ⇒ Idle
Running ⇒ Complete	Producing	Running ⇒ Complete
Pausing ⇒ Paused	Standby	subroutines, breakpoints, or both.
Holding ⇒ Held	Holding ⇒ Held	Holding ⇒ Held
Restarting	none	Restarting
Stopping ⇒ Stopped	Stopping ⇒ Stopped	Stopping ⇒ Stopped
Aborting ⇒ Aborted	Aborting ⇒ Aborted	Aborting ⇒ Aborted

Minimum System Requirements

To develop PhaseManager programs, you need:

- ControlLogix controller with firmware revision 15.0 or later
- communication path to the controller
- RSLogix 5000 software version 15.0 or later

To enable PhaseManager support, you need the full or professional editions of RSLogix 5000 software or the optional PhaseManager add-on (9324-RLDPMENE) to your RSLogix 5000 software package.

Equipment Phase Instructions

The controller supports several instructions to support equipment phases. The instructions are available in ladder diagram (LD) and structured text (ST).

If you want to:	Use this instruction:
signal a phase that the state routine is complete so go to the next state	PSC
change the state or substate of a phase	PCMD
signal a failure for a phase	PFL
clear the failure code of a phase	PCLF
initiate communication with RSBizWare Batch software	PXRQ
clear the NewInputParameters bit of a phase	PRNP
set up breakpoints within the logic of a phase	PPD
take ownership of a phase to either: • prevent another program or RSBizWare Batch software from commanding a phase • make sure another program or RSBizWare Batch software does not already own a phase	PATT
relinquish ownership of a phase	PDET
override a command	POVR

For more information… The PhaseManager User Manual, LOGIX-UM001 provides information on how to design, configure, and program, and phase manager application.

Notes:

Configure Redundancy

Use This Chapter

See:

- *ControlLogix Redundancy System User Manual*, 1756-UM523

The ControlLogix redundancy system uses an identical pair of ControlLogix chassis to keep your machine or process running if a problem occurs with a any equipment in a redundant chassis.

This chapter summarizes:

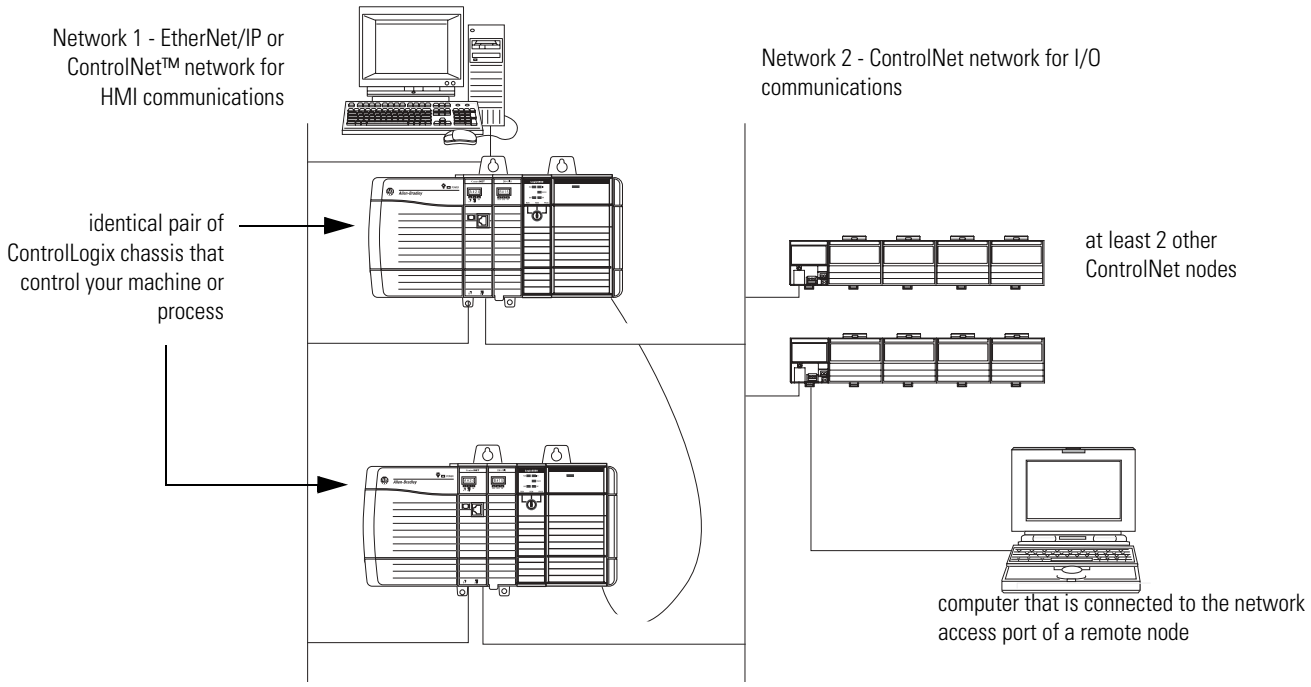
For this information:	See:
ControlLogix Redundancy Overview	9-1
Build a Redundant System	9-3
ControlNet Considerations in Redundant Systems	9-4
EtherNet/IP Considerations in Redundant Systems	9-5
Redundancy and Scan Time	9-6
Minimum System Requirements	9-6

ControlLogix Redundancy Overview

Redundancy provides for higher system availability by switching control to a secondary controller chassis if anything in the primary controller chassis fails. The redundant system switches from primary to secondary upon:

- power loss to primary chassis.
- hardware or firmware failure of any module in primary chassis.
- major fault in the user program on the primary controller.
- disconnection of a ControlNet tap or ControlNet cable break to a 1756-CNB in the primary chassis
- disconnection of an Ethernet patch cable from a 1756-ENBT or 1756-EWEB in the primary chassis.
- removal of any module in the primary chassis.
- user command causing a switchover.

The following diagram shows the layout of a simple redundant set-up.



Redundancy requires no additional programming and is transparent to any devices connected over a EtherNet/IP or ControlNet network. It uses 1757-SRM modules to maintain communication between the pair of redundant chassis.

Depending on how you organize your RSLogix™ 5000 project, outputs may or may not experience a change in state (bump) during a switchover:

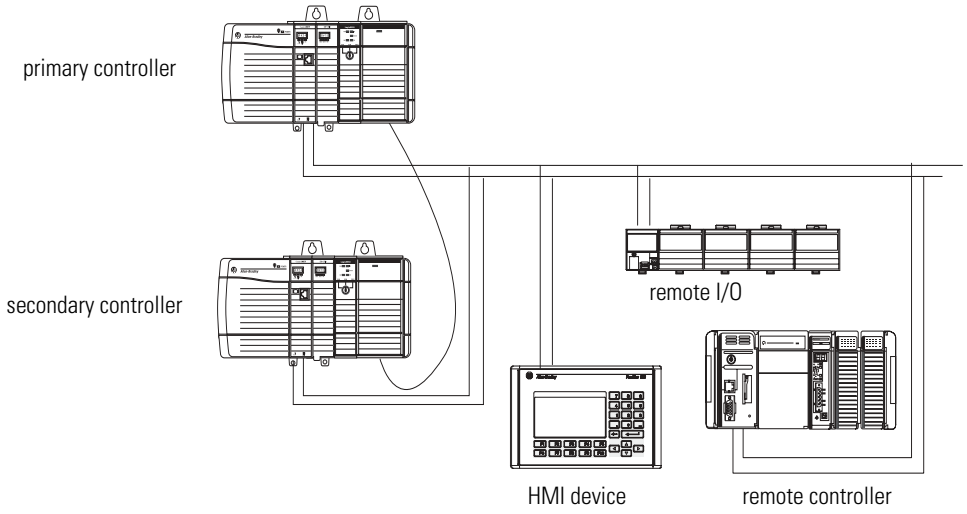
- During the switchover, outputs that are controlled by the highest priority task will experience a bump-less switchover. (i.e., Outputs will not revert to a previous state.)
- Outputs in lower priority tasks *may* experience a change of state.

The switchover time of a redundant system depends on the type of failure and the network update time (NUT) of the ControlNet network. For a NUT of 10 ms, the switchover time is approximately 80 ms to 220 ms.

Build a Redundant System

To build a typical, redundant system:

1. Start with any ControlLogix chassis.
2. Add a 1756-L55, 1756-L61, 1756-L62, or 1756-L63 controller.
3. Add one or more ControlNet (1756-CNB, 1756-CNBR) or EtherNet/IP (1756-ENBT) communication modules.
4. Add one 1757-SRM redundancy module.
5. Set up a second chassis that is identical to the first chassis.
6. Connect the 1757-SRM redundancy modules in both chassis together.
7. Add I/O modules, operator interfaces, and other devices to the ControlNet network.



System considerations

Modules in Primary and Secondary Chassis:	Consideration:
ControlLogix controller	• as of firmware revision 13, you can use these combinations of ControlLogix controllers in a redundant chassis: – one 1756-L55 controller – two 1756-L55 controllers – one 1756-L6x controller • when configured for redundancy, the secondary controller automatically receives and buffers data • a redundant controller uses twice as much data memory and I/O memory space as a non-redundant controller • the controllers synchronize data transfers to support a bumpless switchover: – a redundant controller has a significantly longer scan time than a non-redundant controller – this scan time impact can affect high-speed processes requiring fast (<50 msec) program scans. – minimize the scan time impact by packing data into arrays and structures so the data transfer from primary to secondary controller is more efficient • you do not need a special version of RSLogix 5000 software for redundancy
communication modules	• only controllers, 1756-CNB, -CNBR modules, 1756-ENBT modules, and one 1757-SRM redundancy module can be in a redundant controller chassis • you can have as many as two EtherNet/IP modules in a redundant chassis • you can have as many as five communication modules total in a redundant chassis; for example, two EtherNet/IP modules and three ControlNet modules. • to connect to other networks, bridge through another ControlLogix chassis • use a separate network for HMI and I/O communications
I/O modules	• all I/O is remote from the redundant controller chassis
redundant power supplies	• the 1756-PA75R and 1756-PB75R are redundant power supplies that provide for high availability of chassis power

ControlNet Considerations in Redundant Systems

You can have as many as five ControlNet communication modules in a redundant chassis. You can use 1756-CNB and 1756-CNBR Controlnet Modules.

You must have at least two ControlNet nodes external to the redundant controller chassis to avoid timeouts on switchover.

The lowest ControlNet node must be outside the redundant controller chassis.

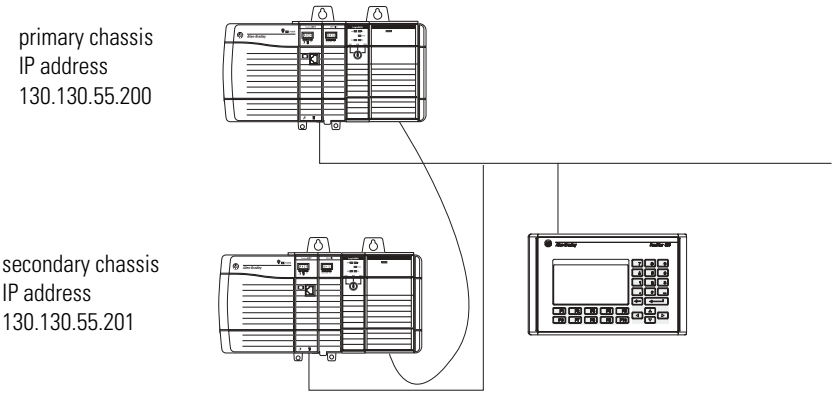
EtherNet/IP Considerations in Redundant Systems

You can have as many as two EtherNet/IP modules in the redundant chassis. You can use 1756-ENBT EtherNet/IP and 1756-EWEB EtherNet/IP Web Server modules.

In a redundant system, use EtherNet/IP for HMI communications or inter-controller messaging only. HMI can talk directly to the primary controller. You no longer need RSLinx Alias Topics. Redundancy does not support EtherNet/IP for I/O control or producing and consuming data.

IP address swapping

Firmware version 13 supports IP address swapping in redundant systems. Configure the primary and secondary EtherNet/IP modules with the same IP address. The primary EtherNet/IP module takes the IP address; the secondary takes that address plus one in the last address segment.



On switchover, the EtherNet/IP modules swap IP addresses. HMI devices automatically continue to communicate with the primary controller. Because of the way EtherNet/IP works, communications between the controller and an HMI device halts for several seconds (typically less than a minute) while the IP address propagates when a switchover occurs.

Use a dedicated ControlNet network instead of an EtherNet/IP network if you need a bumpless HMI connection.

Redundancy and Scan Time

The primary controller stops at the end of every program to crossload fresh data to the secondary controller. This keeps the secondary controller up-to-date and ready to take over. It also increases the scan time when compared to a non-redundant system.

The length of time for the crossload depends on the how much data the primary controller has to crossload:

- The primary controller crossloads any tag to which an instruction wrote a value (even the same value) since the last crossload.
- Crossloading also requires a small amount of overhead time to tell the secondary controller which program the primary controller is executing.

Minimum System Requirements

The following table lists the possible equipment for a ControlLogix redundancy system:

Quantity	Item	Notes
2	ControlLogix chassis	both chassis must be the same size
2	ControlLogix power supply	
2	ControlLogix controller	• use either 1756-L55, 1756-L61, 1756-L62, or 1756-L63 controllers • use the same catalog number and memory size controllers in each chassis
2	ControlLogix ControlNet communication module	use series D modules
2	ControlLogix 10/100 Mbps Ethernet/IP communication module	• optional • can use a second pair of 1756 ControlNet communication modules for HMI/workstation communication
2	1757 system redundancy module	
1	1757 system redundancy cable	standard lengths are available
2	additional ControlNet nodes	• place all I/O in remote chassis or din rails • use a ControlNet network for all I/O • add at least 2 nodes to each ControlNet network in addition to the redundant chassis pair

For more information… The *ControlLogix Redundancy System User Manual*, 1756-UM523 provides information on how to design, install, configure and program, and maintain a ControlLogix redundancy system.

SIL 2 Certification

Use This Appendix

See:

- *Using ControlLogix in SIL 2 Applications Safety Reference Manual*, 1756-RM001

Components of the ControlLogix system are type-approved and certified for use in SIL 2 applications, according to IEC 61508 and AK4 applications according to DIN V19250. SIL requirements are based on the standards current at the time of certification.

This appendix summarizes:

For this information:	See:
SIL 2 Overview	A-1
SIL 2 Application	A-2

IMPORTANT For a list of ControlLogix system components that meet SIL 2 requirements, see *Using ControlLogix in Sil 2 Applications Reference Manual,* publication 1765-RM001

SIL 2 Overview

A Safety Integration Level (SIL) is a numeric designator assigned to a safety system that indicates that system's ability to perform its safety function. The SIL 2 TYPE certification of ControlLogix products by TUV, an internationally-recognized and accredited test laboratory certification center, assures the suitability of ControlLogix products for use in up to a SIL 2 safety application. TUV certification is based primarily on compliance with IEC 61508 Functional Safety of Electrical/Electronic/ Programmable Electronic Safety-Related Systems requirements. It also includes a number of application-independent standards, DIN V 19250 and VDE 0801, application-dependent standards, prEN 50156 for ESD applications, DIN EN 54 for fire and gas application, and environmental and electrical safety standards, IEC 61131-2, EN 50178, EN 50081-2, and EN 61000-2:2000.

These requirements consist of mean time between failures (MTBF), probability of failure, failure rates, diagnostic coverage and safe failure fractions that fulfill SIL2 criteria. The results make the ControlLogix system suitable up to, and including, SIL2. When the ControlLogix system is in the maintenance or programming mode, the user is responsible for maintaining a safe state.

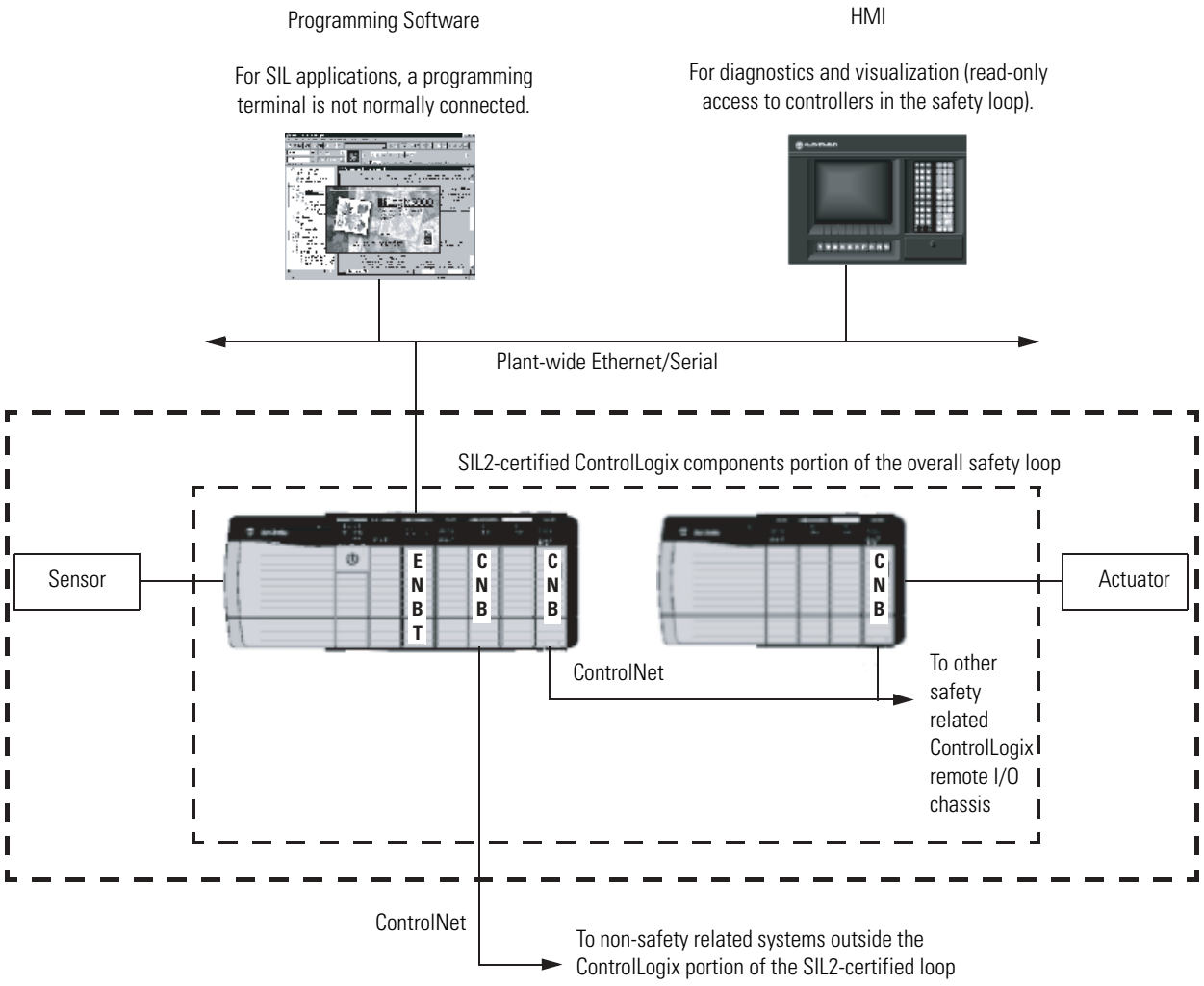
For support in creation of programs, the PADT (Programming and Debugging Tool) is required. The PADT for ControlLogix is RSLogix 5000, per IEC 61131-3, and this Safety Reference Manual.

SIL 2 Application

In obtaining SIL 2 certification for a ControlLogix system, Rockwell Automation did not need to create a special line of products to meet stringent SIL 2 requirements. Sophisticated diagnostics and high levels of reliability are standard design in ControlLogix processors, I/O modules, and communication products. It is that same, standard design that readily provides the reliability needed to achieve SIL 2 certification.

A typical ControlLogix SIL loop, includes:

- the overall safety loop
- the ControlLogix portion of the overall safety loop
- how other devices (for example, HMI) connect to the loop, while operating outside the loop



For more information... The *Using ControlLogix in SIL2 Applications Safety Reference*, 1756-RM001 describes the ControlLogix system components that are approved for use ion SIL2 applications.

Maintain Nonvolatile Memory

Use This Chapter

The 1756-L6x controllers support the 1784-CF64 CompactFlash card for nonvolatile memory. The 1756-L55M22, 1756-M23, and 1756-M24 controllers have built-in nonvolatile memory.

If the controller loses power and does not have enough battery capacity, it loses the project in user memory. Nonvolatile memory lets you keep a copy of your project on the controller. The controller does not need power to keep this copy.

You can load the copy from nonvolatile memory to user memory of the controller:

- on every power-up
- whenever there is no project in the controller and it powers-up
- anytime through RSLogix 5000 software

For details, see:

For this information:	See:
Choose a Controller That Has Nonvolatile Memory	11-2
Use a CompactFlash Reader	11-3

IMPORTANT Nonvolatile memory stores the contents of the user memory at the time that you store the project.

- Changes that you make after you store the project are *not* reflected in nonvolatile memory.
- If you make changes to the project but do not store those changes, you overwrite them when you load the project from nonvolatile memory. If this occurs, you have to upload or download the project to go online.
- If you want to store changes such as online edits, tag values, or a ControlNet network schedule, store the project again after you make the changes.

Choose a Controller That Has Nonvolatile Memory

The following ControlLogix controllers have nonvolatile memory.

Controller Type:	Catalog #:	Firmware Revision:	Requires a 1784-CF64 Industrial CompactFlash memory card:
ControlLogix5555	1756-L55M22	10.x or later	no
	1756-L55M23	8.x or later	no
	1756-L55M24	8.x or later	no
ControlLogix5560M03SE	1756-L60M03SE	13.x or later	yes
ControlLogix5561	1756-L61	12.x or later	yes
ControlLogix5562	1756-L62	12.x or later	yes
ControlLogix5563	1756-L63	11.x or later	yes

Prevent a major fault during a load

If the major and minor revision of the project in nonvolatile memory does not match the major and minor revision of the controller, a major fault *may* occur during a load.

If the controller:	Then:
does not use a CompactFlash card	Make sure that the major and minor revision of the project in nonvolatile memory matches the major and minor revision of the controller. The nonvolatile memory of the controller stores only the project. It does not store the firmware for the controller.
uses a CompactFlash card	The CompactFlash card stores the firmware for projects ≥ 12.0. Depending on the current revision of the controller, you may be able to use the CompactFlash card to update the firmware of the controller and load the project.

Use a CompactFlash Reader

If the revision of the project or projects on your CompactFlash card are ≥ 12, then the card is formatted using the FAT16 file system.

Typically, you do *not* have to manage the files on a CompactFlash card. The card automatically loads the project that you most recently stored. For additional flexibility, the file system also lets you:

- manually change which project loads from the CompactFlash card
- manually change the load parameters for a project

A sample controller project that reads and writes a CompactFlash card is available with RSLogix 5000 Enterprise programming software. From RSlogix 5000 software, select Help → Vendor Sample Projects to display a list of available, sample projects.

For more information... The *Logix5000 Controllers Common Procedures Manual*, 1756-PM001 provides information on how to:

- store a project to nonvolatile memory
- load a project from nonvolatile memory
- use a CompactFlash reader

Notes:

Maintain the Battery

Use This Chapter

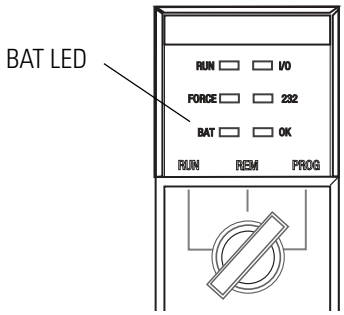
The ControlLogix controller supports these batteries:

With this controller	And this series	Use
ControlLogix5555 ControlLogix5560M03SE	all	1756-BA1 battery **or** 1756-BATM battery module
ControlLogix5561 ControlLogix5562 ControlLogix5563	A	1756-BA1 battery **or** 1756-BATM battery module
	B	1756-BA2 battery

For details, see:

For this information:	See:
Check If the Battery Is Low	12-2
Estimate 1756-BA1 Battery Life (1756-L55Mx all series and 1756-L6x series A controllers)	12-2
Estimate 1756-BA2 Battery Life (1756-L6x series B controllers only)	12-4
Maintain a 1756-BATM Battery Module (1756-L55Mx all series and 1756-L6x series A controllers only)	12-7
Store Batteries	12-8

Check If the Battery Is Low



When the battery is about 95% discharged, the controller gives the following low-battery warnings:

- The BAT LED on the front of the controller turns solid red.
- A minor fault occurs (type 10, code 10).

ATTENTION

To prevent possible battery leakage, even if the BAT LED is off, replace a battery according to the following schedule:

If the temperature 1 in. below the chassis is:	Replace the battery within:
0° to 35° C	No required replacement
36° to 40° C	3 years
41° to 45° C	2 years
46° to 50° C	16 months
51° to 55° C	11 months
56° to 60° C	8 months

Estimate 1756-BA1 Battery Life

(1756-L55Mx all series and 1756-L6x series A controllers)

To estimate how long a 1756-BA1 battery will support controller memory:

1. Determine the temperature (° C) 1 in. below the chassis.
2. Determine the percentage of time that the controller is powered off per week.

EXAMPLE

If a controller is off:

- 8 hr/day during a 5-day work week
- all day Saturday and Sunday

Then the controller is off 52% of the time:

1. total hours per week = 7 x 24 = 168 hours
2. total off hours per week = (5 days x 8 hr/day) + Saturday + Sunday = 88 hours
3. percentage off time = 88/168 = 52%

4. Use Table 12.A or Table 12.B on page 12-3 to determine the estimated worst-case battery life before and after the BAT LED turns on.

5. For each year of battery life, decrease the time before the BAT LED turns on by the percentage that is shown in the table. (Do not decrease the time after the BAT LED turns on.)

IMPORTANT If the BAT LED turns on when you apply power to the controller, the battery life *may* be less then the table below indicates. Some of the battery life may have been used up while the controller was off and unable to turn on the BAT LED.

Table 12.A Worst-case estimates of battery life for the 1756-BA1 battery

Controller:	Temperature:	Time before BAT LED turns on:			Time after BAT LED turns on and then power off 100%:
		Power off 100%	Power off 50%	Yearly decrease:	
1756-L55M12 1756-L55M13	60° C	57 days	110 days	23%	69 hours
	25° C	63 days	123 days	17%	76 hours
	0° C	60 days	118 days	17%	73 hours
1756-L55M14	60° C	29 days	57 days	23%	35 hours
	25° C	30 days	61 days	17%	37 hours
	0° C	24 days	48 days	17%	30 hours
1756-L55M16	60° C	15 days	30 days	23%	18 hours
	25° C	13 days	27 days	17%	16 hours
	0° C	6 days	12 days	36%	7 hours
1756-L55M22 1756-L55M23	Use the values for the 1756-L55M13 controller.				
1756-L55M24	Use the values for the 1756-L55M14 controller.				
1756-L63	60° C	22 days	43 days	23%	6 hours
	25° C	21 days	42 days	17%	28 hours
	0° C	14 days	28 days	17%	2.5 days

Table 12.B Worst-case estimates of life for the 1756-BATA battery

Controller:	Temp:	Time before BAT LED turns on:			Time after BAT LED turns on and then power off 100%:
		Power off 100%	Power off 50%	Yearly decrease:	
1756-L55M12 1756-L55M13	60° C	190 days	396 days	11%	190 days
	25° C	299 days	562 days	5%	299 days
	0° C	268 days	562 days	6%	268 days
1756-L55M14	60° C	130 days	270 days	11%	139 days
	25° C	213 days	391 days	5%	228 days
	0° C	180 days	381 days	6%	193 days
1756-L55M16	60° C	71 days	160 days	13%	76 days
	25° C	133 days	253 days	5%	142 days
	0° C	105 days	220 days	6%	112 days
1756-L55M22 1756-L55M23	Use the values for the 1756-L55M13 controller.				
1756-L55M24	Use the values for the 1756-L55M14 controller.				
1756-L63	60° C	98 days	204 days	11%	104 days
	25° C	146 days	268 days	5%	157 days
	0° C	105 days	222 days	6%	113 days

Estimate 1756-BA2 Battery Life

(1756-L6x series B controllers only)

Use the following table to estimate how long before the battery becomes low (BAT LED = solid red).

Maximum Temperature (° C) 1 in. Below the Chassis.	Power Cycles	Battery Life Before the BAT LED Turns Red (Worst-Case Estimate)			
		Project Size			
		1M bytes	2M bytes	4M bytes	8M bytes
0°…40° C	3 per day	3 years	3 years	26 months	20 months
	2 per day or less	3 years	3 years	3 years	31 months
41°…45° C	3 per day	2 years	2 years	2 years	20 months
	2 per day or less	2 years	2 years	2 years	2 years
46°…50° C	3 per day or less	16 months	16 months	16 months	16 months
51°…55° C	3 per day or less	11 months	11 months	11 months	11 months
56°…60° C	3 per day or less	8 months	8 months	8 months	8 months

EXAMPLE

Under the following conditions…

- The maximum temperature 1 in. below the chassis = 45º C.
- You cycle power to the controller 3 times per day.
- The controller contains an 8M byte project.

…the battery will last at least 20 months before the BAT LED turns red.

Estimate warning time

Use the following table to estimate the battery life after the low-battery warning (BAT LED = solid red). Use these times whether or not the controller has power. There is always a small constant drain on the battery.

IMPORTANT

When you power up the controller, see if there is a low-battery warning. If you get a low-battery warning for the first time, you have less battery life than this table shows. While powered down, the controller still drains the battery but it can't give the low-battery warning.

Maximum Temperature (° C) 1 in. Below the chassis.	Power Cycles	Battery Life After the BAT LED Turns Red (Worst Case)			
		Project Size			
		1M bytes	2M bytes	4M bytes	8M bytes
0°…20° C	3 per day	26 weeks	18 weeks	12 weeks	9 weeks
	1 per Day	26 weeks	26 weeks	26 weeks	22 weeks
	1 per Month	26 weeks	26 weeks	26 weeks	26 weeks
21°…40° C	3 per day	18 weeks	14 weeks	10 weeks	8 weeks
	1 per Day	24 weeks	21 weeks	18 weeks	16 weeks
	1 per Month	26 weeks	26 weeks	26 weeks	26 weeks
41°…45° C	3 per day	12 weeks	10 weeks	7 weeks	6 weeks
	1 per Day	15 weeks	14 weeks	12 weeks	11 weeks
	1 per Month	17 weeks	17 weeks	17 weeks	17 weeks
46°…50° C	3 per day	10 weeks	8 weeks	6 weeks	6 weeks
	1 per Day	12 weeks	11 weeks	10 weeks	9 weeks
	1 per Month	12 weeks	12 weeks	12 weeks	12 weeks
51°…55° C	3 per day	7 weeks	6 weeks	5 weeks	4 weeks
	1 per Day	8 weeks	8 weeks	7 weeks	7 weeks
	1 per Month	8 weeks	8 weeks	8 weeks	8 weeks
56°…60° C	3 per day	5 weeks	5 weeks	4 weeks	4 weeks
	1 per Day	6 weeks	6 weeks	5 weeks	5 weeks
	1 per Month	6 weeks	6 weeks	6 weeks	6 weeks

EXAMPLE

Under the following conditions…

- The maximum temperature 1 in. below the chassis = 45° C.
- You cycle power to the controller 3 times per day.
- The controller contains an 8M byte project.

…the battery is good for another 6 weeks after the BAT LED turns red.

Maintain a 1756-BATM Battery Module
(1756-L55Mx all series and 1756-L6x series A controllers only)

Use the 1756-BATM battery module with any 1756-L55 or 1756-L6x controller. The battery module is highly recommended for the higher-memory controllers:

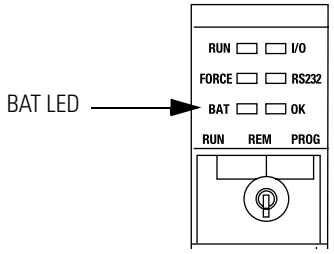
If you have this controller:	And the project is:	Then the 1756-BATM battery module is:
1756-L55M12	→	permitted
1756-L55M13	→	permitted
1756-L55M14	→	highly recommended
1756-L55M16	→	highly recommended
1756-L55M22	stored in nonvolatile memory	not required but permitted
	not stored in nonvolatile memory	permitted
1756-L55M23	stored in nonvolatile memory	not required but permitted
	not stored in nonvolatile memory	permitted
1756-L55M24	stored in nonvolatile memory	not required but permitted
	not stored in nonvolatile memory	highly recommended
1756-L63	stored in nonvolatile memory—requires a 1784-CF64 Industrial CompactFlash card	not required but permitted
	not stored in nonvolatile memory	highly recommended

When the 1756-BATA battery is about 50% discharged, the controller provides the following warnings:

- On the front of the controller, the BAT LED turns on (solid red).
- A minor fault occurs (type 10, code 10).

Check the BAT LED

1. Turn on the chassis power.



2. Is the BAT LED off?

If:	Then:
Yes	The battery module is correctly installed.
No	Go to step 3.

3. Check that the battery module is correctly connected to the controller.

4. Check that the battery assembly is correctly connected to the battery module.

5. If the BAT LED remains on, install another battery assembly (catalog # 1756-BATA).

6. If the BAT LED remains on after you complete step 5, contact your Rockwell Automation representative or local distributor.

Store Batteries

ATTENTION

Follow these general rules to store your batteries:

- Store batteries in a cool, dry environment. We recommend 25°C with 40%…60% relative humidity.
- You may store batteries for up to 30 days between -45°…85°C, such as during transportation.
- To avoid leakage or other hazards, DO NOT store batteries above 60°C for more than 30 days.

For detailed guidelines on how to store batteries, see *Guidelines for Handling Lithium Batteries*, publication AG 5-4. That publication comes with the battery.

Interpret Controller LEDs

RUN Indicator

Color	Description	Recommended Action
off	The controller is in Program or Test mode.	Change the controller mode.
solid green	The controller is in Run mode.	

I/O Indicator

Color	Description	Recommended Action
off	Either: • There are *no* devices in the I/O configuration of the controller. • The controller does not contain a project (controller memory is empty).	 • Add the required devices to the I/O configuration of the controller. • Download the project to the controller.
solid green	The controller is communicating with all the devices in its I/O configuration.	None
flashing green	One or more devices in the I/O configuration of the controller are *not* responding.	Go online with RSLogix 5000 software and check the I/O configuration of the controller.
flashing red	The chassis is bad.	Replace the chassis.

FORCE Indicator

Color	Description	Recommended Action
off	• No tags contain I/O force values. • I/O forces are inactive (disabled).	None
solid amber	• I/O forces are active (enabled). • I/O force values may or may not exist.	USE CAUTION if you install (add) a force. If you install (add) a force, it IMMEDIATELY takes effect.
flashing amber	One or more input or output addresses have been forced to an On or Off state, but the forces have not been enabled.	USE CAUTION if you enable I/O forces. If you enable I/O forces, ALL existing I/O forces also take effect.

RS232 Indicator

Color	Description	Recommended Action
off	There is no activity.	None
solid green	Data is being received or transmitted	None

BAT Indicator

Color	Description		Recommended Action
off	The battery supports memory.		None
solid green	**If the controller is:**	**Then:**	
	series A	The controller does NOT show this indication.	None
	series B	During power-down, the controller is saving the project to its internal nonvolatile memory. If the BAT LED is solid red before you turn off the power, the BAT LED remains solid red even during the save.	None
solid red	Either the battery is:		
	• not installed.		Install a battery.
	• 95% or more discharged		Replace the battery.

OK Indicator

Color	Description		Recommended Action
off	No power is applied.		When ready, turn on power to the controller.
flashing red	**If the controller is:**	**Then:**	
	a new controller (just out of the box)	The controller requires a firmware update.	Update the controller with the correct firmware.
	NOT a new controller (previously in operation)	Major fault occurred.	Clear the fault.
solid red	The controller detected a non-recoverable fault, so it cleared the project from memory.		Clear the fault.
solid green	The controller is OK		None
flashing green	The controller is storing or loading a project to or from nonvolatile memory.		If the controller has a CompactFlash card, leave the card in the controller until the OK LED turns solid green.

Instruction Locator

Where to Find an Instruction

This locator table lists the available instructions, which publications describe the instructions, and which programming languages are available for the instructions.

If the locator lists:	The instruction is documented in:
general	Logix5000 Controllers General Instructions Set Reference Manual, 1756-RM003
process control	Logix5000 Controllers Process Control and Drives Instructions Set Reference Manual, 1756-RM006
motion	Logix5000 Controllers Motion Instructions Set Reference Manual, 1756-RM007
phase	PhaseManager User Manual, LOGIX-UM001

Instruction:	Location:	Languages:
ABL ASCII Test For Buffer Line	general	relay ladder structured text
ABS Absolute Value	general	relay ladder structured text function block
ACB ASCII Chars in Buffer	general	relay ladder structured text
ACL ASCII Clear Buffer	general	relay ladder structured text
ACOS Arc Cosine	general	structured text
ACS Arc Cosine	general	relay ladder function block
ADD Add	general	relay ladder structured text function block
AFI Always False Instruction	general	relay ladder
AHL ASCII Handshake Lines	general	relay ladder structured text
ALM Alarm	process control	structured text function block
AND Bitwise AND	general	relay ladder structured text function block
ARD ASCII Read	general	relay ladder structured text
ARL ASCII Read Line	general	relay ladder structured text

Instruction:	Location:	Languages:
ASIN Arc Sine	general	structured text
ASN Arc Sine	general	relay ladder function block
ATAN Arc Tangent	general	structured text
ATN Arc Tangent	general	relay ladder function block
AVE File Average	general	relay ladder
AWA ASCII Write Append	general	relay ladder structured text
AWT ASCII Write	general	relay ladder structured text
BAND Boolean AND	general	structured text function block
BNOT Boolean NOT	general	structured text function block
BOR Boolean OR	general	structured text function block
BRK Break	general	relay ladder
BSL Bit Shift Left	general	relay ladder
BSR Bit Shift Right	general	relay ladder
BTD Bit Field Distribute	general	relay ladder

Instruction:	Location:	Languages:
BTDT Bit Field Distribute with Target	general	structured text function block
BTR Message	general	relay ladder structured text
BTW Message	general	relay ladder structured text
BXOR Boolean Exclusive OR	general	structured text function block
CLR Clear	general	relay ladder structured text
CMP Compare	general	relay ladder
CONCAT String Concatenate	general	relay ladder structured text
COP Copy File	general	relay ladder structured text
COS Cosine	general	relay ladder structured text function block
CPS Synchronous Copy File	general	relay ladder structured text
CPT Compute	general	relay ladder
CTD Count Down	general	relay ladder
CTU Count Up	general	relay ladder
CTUD Count Up/Down	general	structured text function block
D2SD Discrete 2-State Device	process control	structured text function block
D3SD Discrete 3-State Device	process control	structured text function block
DDT Diagnostic Detect	general	relay ladder
DEDT Deadtime	process control	structured text function block
DEG Degrees	general	relay ladder structured text function block
DELETE String Delete	general	relay ladder structured text
DERV Derivative	process control	structured text function block
DFF D Flip-Flop	process control	structured text function block
DIV Divide	general	relay ladder structured text function block
DTOS DINT to String	general	relay ladder structured text
DTR Data Transitional	general	relay ladder
EOT End of Transition	general	relay ladder structured text
EQU Equal to	general	relay ladder structured text function block
ESEL Enhanced Select	process control	structured text function block
EVENT Trigger Event Task	general	relay ladder structured text
FAL File Arithmetic and Logic	general	relay ladder
FBC File Bit Comparison	general	relay ladder
FFL FIFO Load	general	relay ladder
FFU FIFO Unload	general	relay ladder
FGEN Function Generator	process control	structured text function block
FIND Find String	general	relay ladder structured text
FLL File Fill	general	relay ladder
FOR For	general	relay ladder
FRD Convert to Integer	general	relay ladder function block
FSC File Search and Compare	general	relay ladder
GEQ Greater than or Equal to	general	relay ladder structured text function block
GRT Greater Than	general	relay ladder structured text function block
GSV Get System Value	general	relay ladder structured text
HLL High/Low Limit	process control	structured text function block
HPF High Pass Filter	process control	structured text function block
ICON Input Wire Connector	general	function block
INSERT Insert String	general	relay ladder structured text
INTG Integrator	process control	structured text function block
IOT Immediate Output	general	relay ladder structured text

Instruction:	Location:	Languages:
IREF Input Reference	general	function block
JKFF JK Flip-Flop	process control	structured text function block
JMP Jump to Label	general	relay ladder
JSR Jump to Subroutine	general	relay ladder structured text function block
JXR Jump to External Routine	general	relay ladder
LBL Label	general	relay ladder
LDL2 Second-Order Lead Lag	process control	structured text function block
LDLG Lead-Lag	process control	structured text function block
LEQ Less Than or Equal to	general	relay ladder structured text function block
LES Less Than	general	relay ladder structured text function block
LFL LIFO Load	general	relay ladder
LFU LIFO Unload	general	relay ladder
LIM Limit	general	relay ladder function block
LN Natural Log	general	relay ladder structured text function block
LOG Log Base 10	general	relay ladder structured text function block
LOWER Lower Case	general	relay ladder structured text
LPF Low Pass Filter	process control	structured text function block
MAAT Motion Apply Axis Tuning	motion	relay ladder structured text
MAFR Motion Axis Fault Reset	motion	relay ladder structured text
MAG Motion Axis Gear	motion	relay ladder structured text
MAH Motion Axis Home	motion	relay ladder structured text
MAHD Motion Apply Hookup Diagnostics	motion	relay ladder structured text
MAJ Motion Axis Jog	motion	relay ladder structured text
MAM Motion Axis Move	motion	relay ladder structured text
MAOC Motion Arm Output Cam	motion	relay ladder structured text
MAPC Motion Axis Position Cam	motion	relay ladder structured text
MAR Motion Arm Registration	motion	relay ladder structured text
MAS Motion Axis Stop	motion	relay ladder structured text
MASD Motion Axis Shutdown	motion	relay ladder structured text
MASR Motion Axis Shutdown Reset	motion	relay ladder structured text
MATC Motion Axis Time Cam	motion	relay ladder structured text
MAVE Moving Average	process control	structured text function block
MAW Motion Arm Watch	motion	relay ladder structured text
MAXC Maximum Capture	process control	structured text function block
MCCD Motion Coordinated Change Dynamics	motion	relay ladder structured text
MCCM Motion Coordinated Circular Move	motion	relay ladder structured text
MCCP Motion Calculate Cam Profile	motion	relay ladder structured text
MCD Motion Change Dynamics	motion	relay ladder structured text
MCLM Motion Coordinated Linear Move	motion	relay ladder structured text
MCR Master Control Reset	general	relay ladder
MCS Motion Coordinated Stop	motion	relay ladder structured text
MCSD Motion Coordinated Shutdown	motion	relay ladder structured text
MCSR Motion Coordinated Shutdown Reset	motion	relay ladder structured text
MDF Motion Direct Drive Off	motion	relay ladder structured text
MDO Motion Direct Drive On	motion	relay ladder structured text
MDOC Motion Disarm Output Cam	motion	relay ladder structured text
MDR Motion Disarm Registration	motion	relay ladder structured text

Instruction:	Location:	Languages:
MDW Motion Disarm Watch	motion	relay ladder structured text
MEQ Mask Equal to	general	relay ladder structured text function block
MGS Motion Group Stop	motion	relay ladder structured text
MGSD Motion Group Shutdown	motion	relay ladder structured text
MGSP Motion Group Strobe Position	motion	relay ladder structured text
MGSR Motion Group Shutdown Reset	motion	relay ladder structured text
MID Middle String	general	relay ladder structured text
MINC Minimum Capture	process control	structured text function block
MOD Modulo	general	relay ladder structured text function block
MOV Move	general	relay ladder
MRAT Motion Run Axis Tuning	motion	relay ladder structured text
MRHD Motion Run Hookup Diagnostics	motion	relay ladder structured text
MRP Motion Redefine Position	motion	relay ladder structured text
MSF Motion Servo Off	motion	relay ladder structured text
MSG Message	general	relay ladder structured text
MSO Motion Servo On	motion	relay ladder structured text
MSTD Moving Standard Deviation	process control	structured text function block
MUL Multiply	general	relay ladder structured text function block
MUX Multiplexer	process control	function block
MVM Masked Move	general	relay ladder
MVMT Masked Move with Target	general	structured text function block
NEG Negate	general	relay ladder structured text function block
NEQ Not Equal to	general	relay ladder structured text function block
NOP No Operation	general	relay ladder
NOT Bitwise NOT	general	relay ladder structured text function block
NTCH Notch Filter	process control	structured text function block
OCON Output Wire Connector	general	function block
ONS One Shot	general	relay ladder
OR Bitwise OR	general	relay ladder structured text function block
OREF Output Reference	general	function block
OSF One Shot Falling	general	relay ladder
OSFI One Shot Falling with Input	general	structured text function block
OSR One Shot Rising	general	relay ladder
OSRI One Shot Rising with Input	general	structured text function block
OTE Output Energize	general	relay ladder
OTL Output Latch	general	relay ladder
OTU Output Unlatch	general	relay ladder
PATT Attach to Equipment Phase	phase	relay ladder structured text
PCLF Equipment Phase Clear Failure	phase	relay ladder structured text
PCMD Equipment Phase Command	phase	relay ladder structured text
PDET Detach from Equipment Phase	phase	relay ladder structured text
PFL Equipment Phase Failure	phase	relay ladder structured text
PI Proportional + Integral	process control	structured text function block
PID Proportional Integral Derivative	general	relay ladder structured text
PIDE Enhanced PID	process control	structured text function block
PMUL Pulse Multiplier	process control	structured text function block
PPD Equipment Phase Paused	phase	relay ladder structured text
POSP Position Proportional	process control	structured text function block

Instruction:	Location:	Languages:
PRNP Equipment Phase New Parameters	phase	relay ladder structured text
PSC Phase State Complete	phase	relay ladder structured text
PXRQ Equipment Phase External Request	phase	relay ladder structured text
RAD Radians	general	relay ladder structured text function block
RES Reset	general	relay ladder
RESD Reset Dominant	process control	structured text function block
RET Return	general	relay ladder structured text function block
RLIM Rate Limiter	process control	structured text function block
RMPS Ramp/Soak	process control	structured text function block
RTO Retentive Timer On	general	relay ladder
RTOR Retentive Timer On with Reset	general	structured text function block
RTOS REAL to String	general	relay ladder structured text
SBR Subroutine	general	relay ladder structured text function block
SCL Scale	process control	structured text function block
SCRV S-Curve	process control	structured text function block
SEL Select	process control	function block
SETD Set Dominant	process control	structured text function block
SFP SFC Pause	general	relay ladder structured text
SFR SFC Reset	general	relay ladder structured text
SIN Sine	general	relay ladder structured text function block
SIZE Size In Elements	general	relay ladder structured text
SNEG Selected Negate	process control	structured text function block
SOC Second-Order Controller	process control	structured text function block
SQI Sequencer Input	general	relay ladder
SQL Sequencer Load	general	relay ladder
SQO Sequencer Output	general	relay ladder
SQR Square Root	general	relay ladder function block
SQRT Square Root	general	structured text
SRT File Sort	general	relay ladder structured text
SRTP Split Range Time Proportional	process control	structured text function block
SSUM Selected Summer	process control	structured text function block
SSV Set System Value	general	relay ladder structured text
STD File Standard Deviation	general	relay ladder
STOD String To DINT	general	relay ladder structured text
STOR String To REAL	general	relay ladder structured text
SUB Subtract	general	relay ladder structured text function block
SWPB Swap Byte	general	relay ladder structured text
TAN Tangent	general	relay ladder structured text function block
TND Temporary End	general	relay ladder
TOD Convert to BCD	general	relay ladder function block
TOF Timer Off Delay	general	relay ladder
TOFR Timer Off Delay with Reset	general	structured text function block
TON Timer On Delay	general	relay ladder
TONR Timer On Delay with Reset	general	structured text function block
TOT Totalizer	process control	structured text function block
TRN Truncate	general	relay ladder function block
TRUNC Truncate	general	structured text
UID User Interrupt Disable	general	relay ladder structured text

Instruction:	Location:	Languages:
UIE User Interrupt Enable	general	relay ladder structured text
UPDN Up/Down Accumulator	process control	structured text function block
UPPER Upper Case	general	relay ladder structured text
XIC Examine If Closed	general	relay ladder
XIO Examine If Open	general	relay ladder
XOR Bitwise Exclusive OR	general	relay ladder structured text function block
XPY X to the Power of Y	general	relay ladder structured text function block

Numerics

1756-HYD02
add to controller 7-3
1756-M02AE
add to controller 7-3
1756-M02AS
add to controller 7-3
1756-M03SE
add to controller 7-3
set up 7-5
1756-M08SE
add to controller 7-3
set up 7-5
1756-M16SE
add to controller 7-3
set up 7-5

A

address data 5-8
architecture 1-1
ASCII characters 3-16
axis
add to controller 7-9
check wiring 7-13
get status 7-17
set up 7-10
tune 7-14

B

battery
catalog number C-1
check if low C-2
estimate 1756-BA1 C-2
estimate 1756-BA2 C-4
maintain 1756-BATM C-7
storage C-8
BOOTP 3-3

C

cable, serial 2-1
cache message 4-3
calculate connection use 4-5
catalog number 1-2
change of state 5-3
chassis 5-2
coarse update period
set 7-7
command
give 8-4
communication
ControlNet 3-5
determine timeout with any device 6-7
determine timeout with I/O module 6-8
DeviceNet 3-8
DH+ 3-20
DH-485 3-17
EtherNet/IP 3-3
format 5-3
FOUNDATION Fieldbus 3-24
HART 3-25
serial 3-10
universal remote I/O 3-21
CompactFlash
for more information B-3
load considerations B-2
overview B-1
reader B-3
supported controller B-2
configuration folder 5-2
configure
ControlNet I/O module 5-6
DeviceNet I/O module 5-7
EtherNet/IP I/O module 5-5
I/O module 5-2
SERCOS interface module 7-5
serial driver 2-3
connect
ControlNet 3-5
DeviceNet 3-8
DH+ 3-20
DH-485 3-17
EtherNet/IP 3-3
FOUNDATION Fieldbus 3-24
HART 3-25
RIO 3-21
serial 2-1, 3-10
connection
calculate use 4-5
consume data 4-1
ControlNet 3-7
determine timeout with any device 6-7
determine timeout with I/O module 6-8
DeviceNet 3-9
EtherNet/IP 3-4
example 4-6
for more information 4-4
I/O module 5-4
message 4-3
monitor 6-7
overview 4-1
produce data 4-1
summary 4-4

consume data
connection use 4-1
for more information 4-2
overview 3-1
continuous task 6-2
control distributed I/O
overview 3-1
controller
battery module C-7
catalog number 1-2
check battery C-2
CompactFlash B-2
consume data 3-1
control distributed I/O 3-1
CPU 1-3
design 1-3
estimate battery life C-2, C-4
fault handler 6-9
install 1-4
message 3-1
monitor status 6-6
non-volatile memory B-2
path 2-5
produce data 3-1
redundancy 9-1
serial connection 2-1
status 6-6
ControlNet
connection use 3-7
distributed I/O 5-6
example configuration 3-6
for more information 3-7
module capability 3-5
overview 3-5
redundancy considerations 9-4
scheduled 3-7
unscheduled 3-7
coordinate system
overview 7-17
coordinated system time master
set 7-2
COS 5-3
CPU 1-3
CST master
See coordinated system time master

D

design 1-3
develop application
fault handler 6-9
for more information 6-3
monitor connection 6-7
monitor status 6-6
overview 6-1
program 6-2
programming language 6-5
routine 6-2
tag 6-4
task 6-1
DeviceNet
connection use 3-9
distributed I/O 5-7
example configuration 3-9
for more information 3-10
module capability 3-9
overview 3-8
DF1 configuration 3-11
DF1 device 3-12
DH+
example configuration 3-20
for more information 3-21
module capability 3-21
overview 3-20
DH-485
controller configuration 3-19
overview 3-17
DHCP 3-3
direct connection 5-4
distributed I/O
ControlNet 5-6
DeviceNet 5-7
EtherNet/IP 5-5
overview 3-1
drive
add SERCOS interface drive 7-4
check wiring 7-13

E

electronic keying 5-3
equipment phase
compared to PackML 8-6
compared to S88 8-6
instructions 8-1
monitor 8-6
overview 8-1
equipment phase instructions
overview 8-1

EtherNet/IP
connection use 3-4
distributed I/O 5-5
example configuration 3-4
for more information 3-5
module capability 3-3
overview 3-3
redundancy considerations 9-5
event task 6-2
example system 1-1

F

fault handler 6-9
faults
axis 7-18
motion control 7-18
FBD 6-5
FOUNDATION Fieldbus 3-24
function block diagram 6-5

G

GSV instruction 6-6

H

HART 3-25
Highway Addressable Remote Transducer, see HART 3-25
hookup tests
run 7-13

I

I/O
address data 5-8
chassis 5-2
communication format 5-3
configuration folder 5-2
configure 5-1
connection use 5-4
COS 5-3
determine update 5-11
direct connection 5-4
distributed via ControlNet 5-6
distributed via DeviceNet 5-7
distributed via EtherNet/IP 5-5
electronic keying 5-3
for more information 5-4
module capability 5-1
monitor 5-1
monitor connection 6-8
I/O (continued)
place 5-1
rack-optimized 5-4
reconfigure module 5-12
RPI 5-3
install 1-4
instruction locator B-1

L

ladder diagram 6-5
language 6-5
locator B-1
Logix5000 controller environment 1-1

M

master mode 3-11
message
cache 4-3
connection use 4-3
for more information 4-4
overview 3-1
reconfigure I/O module 5-13
Modbus support 3-17
motion control
add axis 7-9
choose a motion module 7-3
coarse update period 7-7
coordinate system 7-17
execution 7-7
handle faults 7-18
overview 7-1
program 7-15
set the coordinated system time master 7-2
set up an axis 7-10
status information 7-17
motion group
set up 7-7
motion instructions
overview 7-15
motion planner
set period 7-7

N

network
overview 3-1
non-volatile memory
for more information B-3
load considerations B-2
overview B-1
supported controller B-2

P

periodic task 6-2
phase
See equipment phase
point-to-point 3-11
produce data
connection use 4-1
for more information 4-2
overview 3-1
program 6-2
programming language 6-5

R

rack-optimized connection 5-4
reconfigure I/O module 5-12
redundancy
considerations 9-4
ControlNet 9-4
EtherNet/IP 9-5
example system 9-2
for more information 9-6
overview 9-1
requirements 9-3
switchover 9-2
relay ladder 6-5
requested packet interval 5-3
resource 1-3
RIO, see universal remote I/O 3-21
routine 6-2
RPI 5-3
RS-232 DF1 Device driver 2-3

S

safety integration level, see SIL 2 A-1
scheduled 3-7
sequential function chart 6-5
SERCOS interface drive
add to controller 7-4
SERCOS interface module
set up 7-5
SERCOS interface modules
choose 7-3
serial
cable 2-1
communicate with ASCII device
ASCII device 3-14
communicate with DF1 device 3-12
controller communication 3-10
controller connection 2-1
DH-485 configuration 3-17, 3-19
driver 2-3
for more information 3-13, 3-17
Modbus support 3-17
port configuration 3-11
select controller path 2-5
SFC 6-5
SIL 2 certification
example application A-2
for more information A-2
overview A-1
slave mode 3-11
SSV instruction 6-6
ST 6-5
start 1-1
state model
See states
states
compared to PackML 8-6
compared to S88 8-6
manually step through 8-6
overview 8-3
tranisition 8-4
status 6-6
structured text 6-5
system layout 1-1

T

tag
for more information 6-4
organize 6-4
task 6-1
tune
axis 7-14

U

universal remote I/O
for more information 3-23
module capability 3-23
overview 3-21
unscheduled 3-7
update 5-11

W

where to start 1-1

Notes:

How Are We Doing?

Your comments on our technical publications will help us serve you better in the future. Thank you for taking the time to provide us feedback.

You can complete this form and mail (or fax) it back to us or email us at RADocumentComments@ra.rockwell.com

Pub. Title/Type ControlLogix Controllers Firmware Revsion 15 User Manual

Cat. No. 1756-L55Mx, 1756-L6x Pub. No. 1756-UM001F-EN-P Pub. Date May 2005 Part No. 957955-24

Please complete the sections below. Where applicable, rank the feature (1=needs improvement, 2=satisfactory, and 3=outstanding).

Overall Usefulness 1 2 3	How can we make this publication more useful for you?	
Completeness (all necessary information is provided) 1 2 3	Can we add more information to help you?	
	procedure/step	illustration feature
	example	guideline other
	explanation	definition
Technical Accuracy (all provided information is correct) 1 2 3	Can we be more accurate?	
	text	illustration
Clarity (all provided information is easy to understand) 1 2 3	How can we make things clearer?	
Other Comments	You can add additional comments on the back of this form.	

Your Name _____

Your Title/Function _____

Location/Phone _____

Would you like us to contact you regarding your comments?

___No, there is no need to contact me

___Yes, please call me

___Yes, please email me at _____

___Yes, please contact me via _____

Return this form to: Rockwell Automation Technical Communications, 1 Allen-Bradley Dr., Mayfield Hts., OH 44124-9705

Fax: 440-646-3525 Email: RADocumentComments@ra.rockwell.com

PLEASE FASTEN HERE (DO NOT STAPLE)

Other Comments

PLEASE FOLD HERE

PLEASE REMOVE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST-CLASS MAIL PERMIT NO. 18235 CLEVELAND OH

POSTAGE WILL BE PAID BY THE ADDRESSEE

Allen-Bradley
RELIANCE ELECTRIC
DODGE
ROCKWELL SOFTWARE
Rockwell Automation

1 ALLEN-BRADLEY DR
MAYFIELD HEIGHTS OH 44124-9705

ASCII Character Codes

Character	Dec	Hex	Character	Dec	Hex	Character	Dec	Hex	Character	Dec	Hex
[ctrl-@] NUL	0	$00	SPACE	32	$20	@	64	$40	‘	96	$60
[ctrl-A] SOH	1	$01	!	33	$21	A	65	$41	a	97	$61
[ctrl-B] STX	2	$02	“	34	$22	B	66	$42	b	98	$62
[ctrl-C] ETX	3	$03	#	35	$23	C	67	$43	c	99	$63
[ctrl-D] EOT	4	$04	$	36	$24	D	68	$44	d	100	$64
[ctrl-E] ENQ	5	$05	%	37	$25	E	69	$45	e	101	$65
[ctrl-F] ACK	6	$06	&	38	$26	F	70	$46	f	102	$66
[ctrl-G] BEL	7	$07	‘	39	$27	G	71	$47	g	103	$67
[ctrl-H] BS	8	$08	(	40	$28	H	72	$48	h	104	$68
[ctrl-I] HT	9	$09	)	41	$29	I	73	$49	i	105	$69
[ctrl-J] LF	10	$I ($0A)	*	42	$2A	J	74	$4A	j	106	$6A
[ctrl-K] VT	11	$0B	+	43	$2B	K	75	$4B	k	107	$6B
[ctrl-L] FF	12	$0C	,	44	$2C	L	76	$4C	l	108	$6C
[ctrl-M] CR	13	$r ($0D)	-	45	$2D	M	77	$4D	m	109	$6D
[ctrl-N] SO	14	$0E	.	46	$2E	N	78	$4E	n	110	$6E
[ctrl-O] SI	15	$0F	/	47	$2F	O	79	$4F	o	111	$6F
[ctrl-P] DLE	16	$10	0	48	$30	P	80	$50	p	112	$70
[ctrl-Q] DC1	17	$11	1	49	$31	Q	81	$51	q	113	$71
[ctrl-R] DC2	18	$12	2	50	$32	R	82	$52	r	114	$72
[ctrl-S] DC3	19	$13	3	51	$33	S	83	$53	s	115	$73
[ctrl-T] DC4	20	$14	4	52	$34	T	84	$54	t	116	$74
[ctrl-U] NAK	21	$15	5	53	$35	U	85	$55	u	117	$75
[ctrl-V] SYN	22	$16	6	54	$36	V	86	$56	v	118	$76
[ctrl-W] ETB	23	$17	7	55	$37	W	87	$57	w	119	$77
[ctrl-X] CAN	24	$18	8	56	$38	X	88	$58	x	120	$78
[ctrl-Y] EM	25	$19	9	57	$39	Y	89	$59	y	121	$79
[ctrl-Z] SUB	26	$1A	:	58	$3A	Z	90	$5A	z	122	$7A
ctrl-[ESC	27	$1B	;	59	$3B	[	91	$5B	{	123	$7B
[ctrl-\] FS	28	$1C	<	60	$3C	\	92	$5C	|	124	$7C
ctrl-] GS	29	$1D	=	61	$3D	]	93	$5D	}	125	$7D
[ctrl-^] RS	30	$1E	>	62	$3E	^	94	$5E	~	126	$7E
[ctrl-_] US	31	$1F	?	63	$3F	_	95	$5F	DEL	127	$7F

Rockwell Automation Support

Rockwell Automation provides technical information on the web to assist you in using our products. At http://support.rockwellautomation.com, you can find technical manuals, a knowledge base of FAQs, technical and application notes, sample code and links to software service packs, and a MySupport feature that you can customize to make the best use of these tools.

For an additional level of technical phone support for installation, configuration and troubleshooting, we offer TechConnect Support programs. For more information, contact your local distributor or Rockwell Automation representative, or visit http://support.rockwellautomation.com.

Installation Assistance

If you experience a problem with a hardware module within the first 24 hours of installation, please review the information that's contained in this manual. You can also contact a special Customer Support number for initial help in getting your module up and running:

United States	1.440.646.3223 Monday – Friday, 8am – 5pm EST
Outside United States	Please contact your local Rockwell Automation representative for any technical support issues.

New Product Satisfaction Return

Rockwell tests all of our products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned:

United States	Contact your distributor. You must provide a Customer Support case number (see phone number above to obtain one) to your distributor in order to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for return procedure.

www.rockwellautomation.com

Corporate Headquarters
Rockwell Automation, 777 East Wisconsin Avenue, Suite 1400, Milwaukee, WI, 53202-5302 USA, Tel: (1) 414.212.5200, Fax: (1) 414.212.5201

Headquarters for Allen-Bradley Products, Rockwell Software Products and Global Manufacturing Solutions
Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe: Rockwell Automation SA/NV, Vorstlaan/Boulevard du Souverain 36-BP 3A/B, 1170 Brussels, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Headquarters for Dodge and Reliance Electric Products
Americas: Rockwell Automation, 6040 Ponders Court, Greenville, SC 29615-4617 USA, Tel: (1) 864.297.4800, Fax: (1) 864.281.2433
Europe: Rockwell Automation, Brühlstraße 22, D-74834 Elztal-Dallau, Germany, Tel: (49) 6261 9410, Fax: (49) 6261 17741
Asia Pacific: Rockwell Automation, 55 Newton Road, #11-01/02 Revenue House, Singapore 307987, Tel: (65) 351 6723, Fax: (65) 355 1733

Publication 1756-UM001F-EN-P - May 2005 PN 957955-24
Supersedes Publication 1756-UM001E-EN-P - August 2002
Copyright © 2005 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A.